ファミコン大作戦 ファミコンソフト集 001

誠文堂新光社

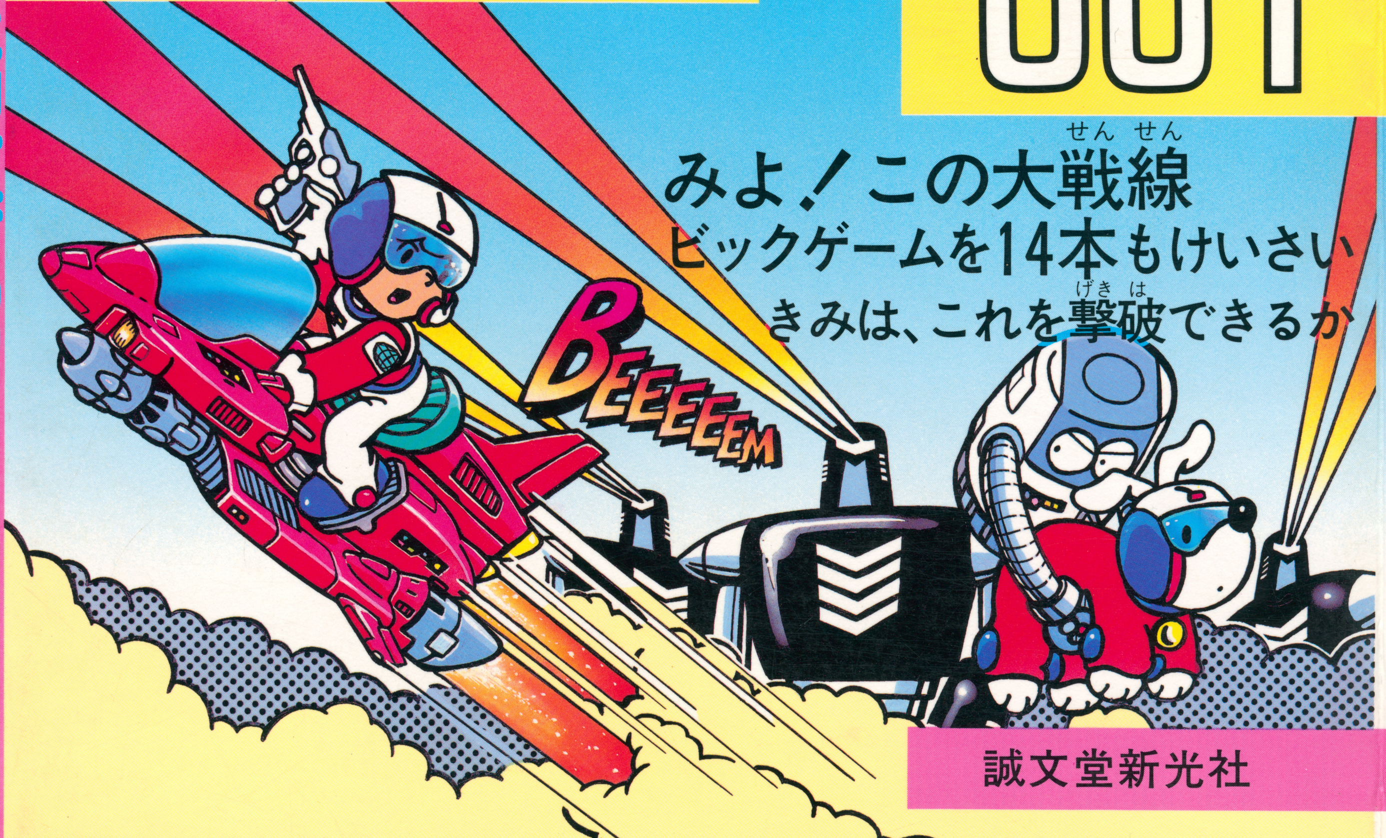
ファミコン大作戦

ファミコンソフト集

ゲーム
14本を公開

001

地球防衛軍 編著
司令 岩田 裕
谷岡康則

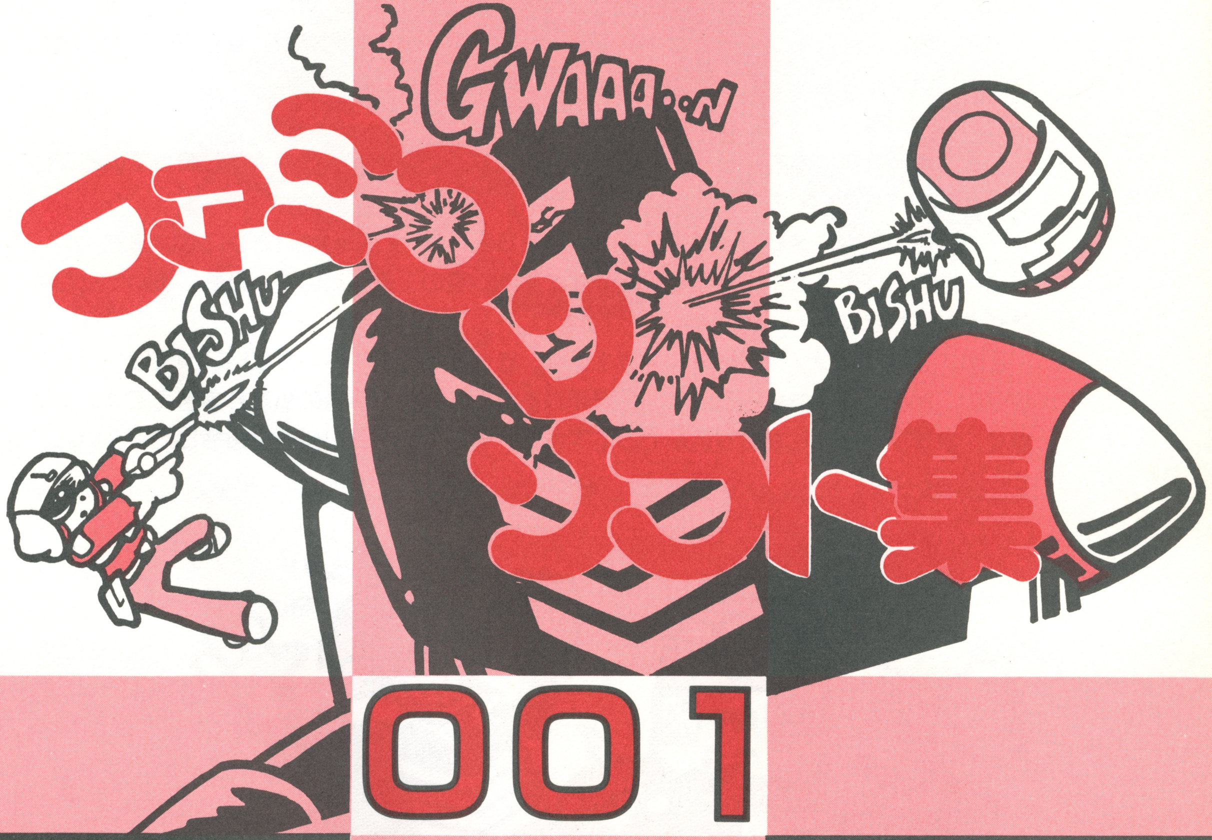


みよ！この大戦線
ビックゲームを14本もけいさい
きみは、これを撃破できるか

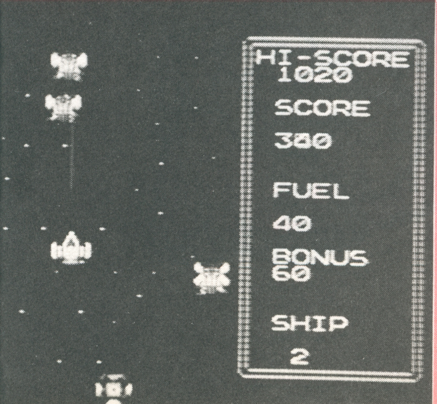
誠文堂新光社

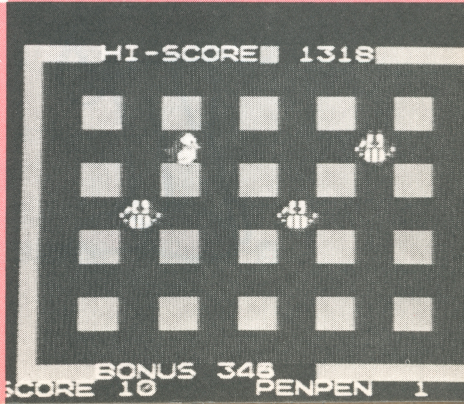
ファミコン大作戦

ファミコンソフト集



001





はじめに

君は，ファミコンを手にして，もうどれくらいたちますか？　ゲームカセットは，何本くらい持っていますか？　そのゲームカセットで遊ぶことだけで，満足していますか？

お店で売られているゲームカセットで遊ぶのも，楽しいことですが，君自身の手で，君だけの新しいゲームを作ってみたいと思ったことはありませんか？

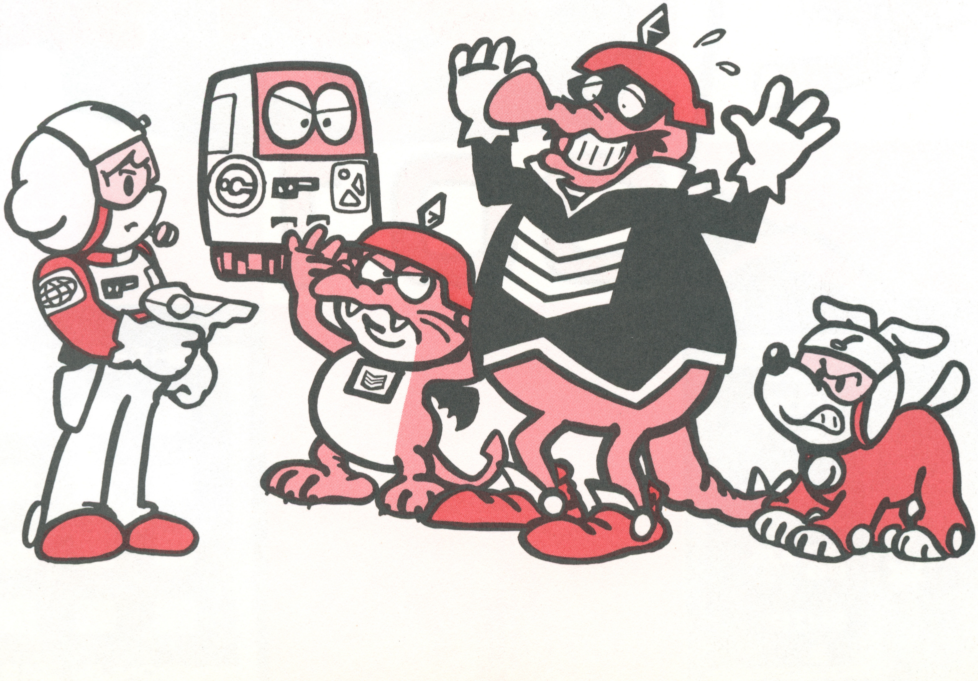
「だって，ベーシックは難しいでしょ!?」

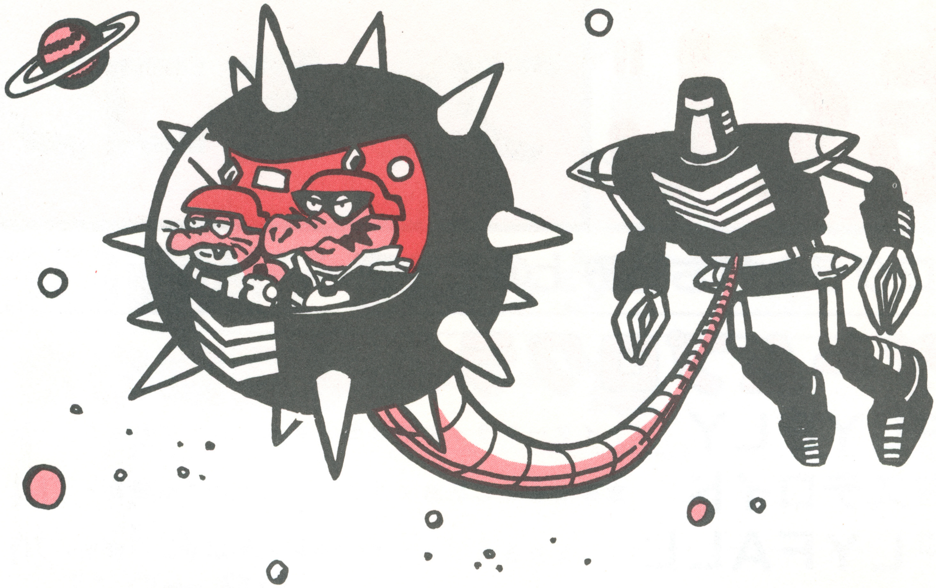
いいえ，そんな事はありません。この「ファミコン　ソフト集　ØØ1」に載っているプログラムを，打ち込んでみて下さい。

「ファミコンは，メモリーが少なくて，おもしろいゲームなんかできっこないヨ!!」

本当にそうでしょうか？　メモリーが少なくても，君の工夫しだいで，おもしろいゲームはたくさん作れます。

この「ファミコン　ソフト集　ØØ1」は，たった1982バイトしかないファミリーベーシックで，どれくらいおもしろいゲームが作れるのか，という事への私たちの答えです。とにかく，まずプログラムを打ち込んでみて下さい。そして，「RUN」と命令して下さい。こ





んなにいろいろなゲームが，こんなに短いプログラムでできるのか，とびっくりされることでしょう。

プログラムを入力していると，きっと
「あれ，ここでなぜこんな事をしているんだろう？」
と思うこともあるでしょう。そうなれば，しめたものです。それは，君のプログラム能力が上達している証拠だからです。

ファミコンは，私たちの友達です。ゲームカセットで遊ぶだけではなく，もっともっと，ファミコンの可能性を引き出してあげて下さい。

「ファミコン　ソフト集」は，この「ØØ1」のあと，「ØØ2」「ØØ3」……と続きます。ファミリーベーシックVer.3を使った大型ゲームも登場してきます。

ファミリーベーシックの入門書「ファミリーベーシック　ホップ」「ステップ」「ジャンプ」と共に，ファミコンと君のそばにおいて下さい。

君と，ファミコンと，私たちとで，君の新しい可能性を開きましょう。

1985年5月
地球防衛軍　スタッフ一同

もくじ ファミコン

プログラム打ち込み上のご注意

ゲームプログラム集

ソフト集 001

BG画面テクニック

BGグラフィック
入力法

BG画面データー集

巻末資料

●ゲームキャラクター表
●プログラムが動かない時は？

●制作・企画　地球防衛軍　岩田裕・谷岡康則・鈴木康夫・真藤美紀子
簑藤桂・北村順・初瀬籠・美崎靖之
●編集・企画　リポート社
●カバー・本文イラスト　高橋裕之（ファクトリー）

© 1985 Seibundo Shinkosha Publishing Co.,Ltd.　※本誌に掲載の記事は，無断転用を禁じます。

このソフト集のプログラム打ち込み上のご注意

プログラムリストは，ファミコン上で動作しているものを，そのままプリントアウトしたものです。

みなさんも知っている通り，ファミリーベーシック　Ver.2は，フリーエリアが2キロ弱しかありません。そのため，ゲームを作るのに非常に苦労しています。

本当はベーシックの命令は，できるだけスペース（空白）をあける方が見やすいのですが，スペース一字分でメモリーも1バイト減ってしまいます。それで

```
GOTO 200
PRINT "TEST"
```

などという行は

```
GOTO200
PRINT"TEST"
```

というようにスペースを詰めているプログラムがあります。こういうプログラムの場合，リスト通りに打ち込まないと

```
?OM ERROR
```

というエラーを起こします。これは，メモリーが足りないという意味のエラーです。

また

```
IFA=1ERA0
IFA=1RETURN
```

というような行が出てきますが，これは

```
IF A=1 THEN ERA 0
IF A=1 THEN RETURN
```

というのが正しい書式（プログラムの書き方）なのですが，実はファミリーベーシックでは「THEN」を省略しても，条件式の後の命令を実行してしまうのです。（他のパソコンでは，こうはうまくいきません）

```
IFA=1ORB=1PRINTA
```

というようにAND，ORで条件式を結んでも大丈夫です。こうすると，プログラムの長さ



が１バイト減るわけです。ふつう，他のパソコンでプログラムを作る時にはおすすめできませんが，このソフト集のプログラムを打ち込む時は，リスト通りにして下さい。

また，この他にも

　　ＩＦＭＯＶＥ（∅）～

　　Ｘ＝ＣＸ－（Ｓ＝２）＊２～

というような書き方をしている所が，いくつかあります。

これは，たとえば

　　Ａ＝２：Ｂ＝（Ａ＝２）：ＰＲＩＮＴ　Ｂ

　　Ａ＝２：Ｂ＝（Ａ＝１）：ＰＲＩＮＴ　Ｂ

とそれぞれ，ダイレクトでやってみて下さい。

カッコ（　）の中が正しい時は「－１」，まちがっている時は「∅」か「Ｂ」に代入されているのが，わかると思います。それを利用した書き方です。そして，

　　ＩＦ　～　ＴＨＥＮ

という文型では，「～」が正しければ「ＴＨＥＮ」以下の命令を実行するわけですが，実は「～」の中が正しいという事は，「～」が「－１」であるという事なのです。そこで

　　ＩＦ　Ａ　ＴＨＥＮ　～

という書き方をすると，「Ａ＝－１」の時「ＴＨＥＮ」以下を実行し，そうでない時は実行しないという意味なのです。

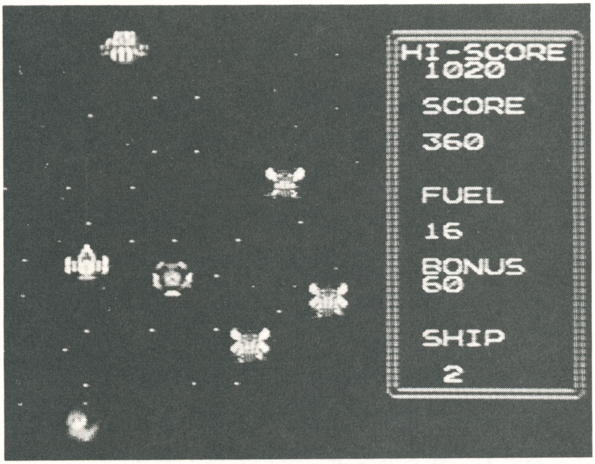
両方とも，プログラムの作り方としては正しいのですが，あとでプログラムを見直す時わかりずらいので，あまりおすすめできませんが，メモリーの節約には有効な手段です。

なお，プログラムリストのプリントアウトの都合上，このソフト集のプログラムリストでは，「ば」や「ぱ」は「は ゛」「は ゜」と２字分のスペースになっていますが，「ば」「ぱ」と１字で入力して下さい（この２字と１字の違いについては，「ファミリーベーシック　ジャンプ」編で説明したいと思います）。

プログラムの入力（打ち込み方）や修整のやり方については，「ファミリーベーシック　ホップ」を参照して下さい。

FLY FLY





人間が生存可能な環境を持つ小惑星“S3612-09”。平和そうに見えたこの世界に思いがけない危険が隠れていた。地球からの調査隊が地球のハエに似た生物「ファイターフライ」の攻撃を受けているらしい。調査隊を救い，基地を作るための増援として，地球から「スターシップ」が発進した。ところが小惑星が近づいた時，ファイターフライの攻撃を受けてしまった。ブンブン飛んで来て体当りをかけてくるとは…。ファイターフライがぶつかるたびに「スターシップ」の燃料が減っていく。このままでは，着陸に必要な燃料まで失ってしまう……。オッ，あれは小惑星からうち出してくれた燃料ユニットだ。やれやれ。

うわっ，隕石だ!! あんなのにぶつかったらおしまいだ……。

★遊び方

プログラムをRUNさせると

PUSH START BUTTON

と表示が出ますから，コントローラーIのSTARTボタンを押してください。

ゲームが始まると「スターシップ」が表示されます。これをコントローラーIの十字ボタンで左右に動かします。宇宙空間ですから，動くためには移動用のロケットエンジンを使います。これを1回使うと（キーを1回押すと），FUELが1減ります。さらに一度動くと，逆

プログラムリスト　(BG画面データーは95ページ)

```
100 CLS:CGEN2:CGSET1:SPRITEON:VIEW
110 PALET S 2,47,48,21,18
120 PALET S 3,47,48,39,22
130 DEFSPRITE0,(0,1,0,0,0)="ピプペポ"
140 DEFSPRITE1,(0,1,0,0,0)=CHR$(180)+CHR$(181)+CHR
$(182)+CHR$(183)
150 FL=10:SP=200:FI=50:NI=150
300 LOCATE0,10:PRINT"PUSH START BUTTON!"
310 IF STRIG(0)<>1GOTO310
320 SC=0:SH=2:B=2500
400 V=0:FU=300:CX=128:VIEW
410 GOSUB3000:GOSUB3100:GOSUB3200:GOSUB3300:SPRITE
1
420 FORI=0TO5:POSITIONI,0,0:NEXT:POSITION6,255,255

500 IF STRIG(0)=8GOSUB1000
510 S=STICK(0):IF FU=0 GOTO540
520 IF S=1 THEN V=V+2:FU=FU-1:GOTO540
530 IF S=2 THEN V=V-2:FU=FU-1
540 CX=CX+V:GOSUB3200
550 IF CX<16 THEN CX=16
560 IF CX>144 THEN CX=144
570 SPRITE0,CX,120
580 IF XPOS(6)<255 AND MOVE(6)=0 THEN ERA6:POSITIO
N6,255,255
590 FOR I=0 TO 5
600 IF ABS(XPOS(I)-CX)<16 AND ABS(YPOS(I)-120)<16
GOSUB2500
```

方向にエンジンを噴射しない限り，いつまでも止まりません。したがって，あまり激しく動くとすぐ燃料がなくなってしまい，姿勢制御が不可能になってしまいます。

敵を攻撃するためには，コントローラーのAボタンを押すとタマを発射できます。

★画面に現われるキャラクター

●ファイターフライ

画面の上から下へ，まっすぐ飛んできます。スターシップにぶつかるとFUELが10減ります。撃ちおとすと10点です。

●ニタニタ

画面の右上から左端まで，横に移動します。敵の偵察部隊で，撃つと10～100点です。

●スピナー

味方の燃料ユニットで，回収する（ぶつかればいい）とFUELが50増えます。

●ファイアーボール

隕石で，画面下から上へ移動します。これにぶつかるとスターシップは，やられてしまいます。破壊できません。

スターシップは3隻，燃料は300です。得点2500点ごとにスターシップが1隻増えます。

★プログラムの説明

100～　初期設定
500～　メインルーチン
1000～　タマ発射
2000～　各キャラクター出現，移動
2500～　衝突処理
3000～　得点，FUEL などの表示
3500～　GAME　OVER

プログラムは，大きく分けると上のようになります。メインルーチンでは，キー入力，スターシップの移動，衝突判定を行なっています。キー入力を調べ（510行），それによってその回の移動量を決め（520～530行），スターシップを移動させます（570行）。

毎回の移動量は変数Vに入っていますから，移動を遅く(反応をにぶく)したい時は540行の，

```
CX=CX+V
```

のVを，適当な数で割ってやります。

衝突の判定は座標の比較で行ない，衝突が起こった時は2500行からのサブルーチンに飛び，ここで衝突した物の種類に応じた処理を行ないます。

```
610 IF XPOS(I)-6<XPOS(6) AND XPOS(I)+14>XPOS(6) AN
D YPOS(I)-8<YPOS(6) AND YPOS(I)+16>YPOS(6) GOSUB15
00
620 IF XPOS(I)=0 GOSUB2000:GOTO640
630 IF NOT MOVE(I) THEN ERAI:POSITIONI,0,0
640 NEXT:GOTO500
1000 IF MOVE(6) THEN RETURN
1010 DEFMOVE(6)=SPRITE(12,1,1,51,0,0):POSITION6,CX
+4,112
1020 MOVE6:RETURN
1500 IF I>3 THEN RETURN
1510 IF I>0 GOTO1530
1520 J=RND(6)*10+50:LOCATE19,14:PRINTJ;" ":SC=(SC+
J-10)MOD32760
1530 ERAI,6:POSITIONI,0,0:POSITION6,255,255
1540 SC=(SC+10)MOD32760:GOSUB3000
1550 IF SC>=B THEN SH=SH+1:GOSUB3300:B=(B+2500)MOD
32500
1560 RETURN
2000 ONI+1GOTO2100,2010,2010,2010,2040,2070
2010 IFRND(FL)>0 THEN RETURN
2020 DEFMOVE(I)=SPRITE(2,5,1,88,0,2):POSITIONI,RND
(129)+16,24
2030 MOVEI:RETURN
2040 IFRND(SP)>0THEN RETURN
2050 DEFMOVE(I)=SPRITE(7,5,3,88,0,1):POSITIONI,RND
(128)+16,24
2060 MOVEI:RETURN
2070 IFRND(FI)>0 THEN RETURN
```

あるキャラクターが移動していない（画面に現われていない）場合には2000行に飛んで，乱数で出現を決定します。出現の確率は各キャラクターによって異なり，それぞれの変数にしまわれています(変数表参照)。処理の開始行は，ファイターフライ(2010行)，スピナー(2040行)，ファイアーボール（2070行)，ニタニタ（2100行）です。

撃ったタマが何かに当った時は1500行に飛び，対象がファイターフライかニタニタなら，そのキャラクターを消し，得点を加算します。ニタニタの場合は得点は乱数で決めます(1520行)。

★プログラムの改造

このゲームの場合，改造といっても変数の値を変えるくらいだと思います。ゲームバランスをくずさないように修正できるのは，次の２つでしょう。

① スターシップの数　320行　SH
② キャラクターの出現度数　150行　変数表参照

また，スターシップ以外の移動スピードは，各スプライトのDEF MOVE文を変えればいいでしょう（表参照)。

★変数表

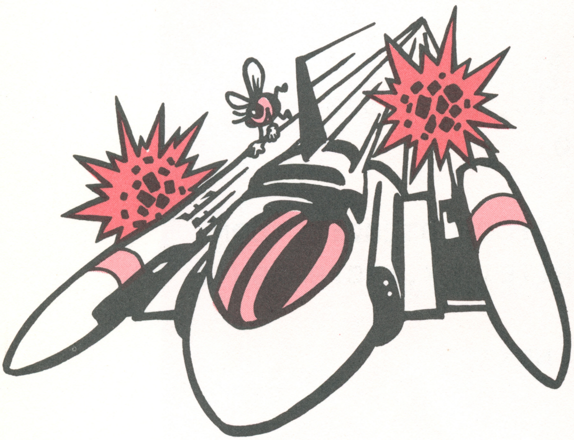
CX	スターシップのX座標	
V	スターシップの移動量	
SH	スターシップの数	
FL	ファイターフライ出現度数	値が小さい程，現われやすくなる。
SP	スピナー出現度数	
FI	ファイアーボール出現度数	
NI	ニタニタ出現度数	
SC	スコア	
HS	ハイスコア	
FU	残り燃料	

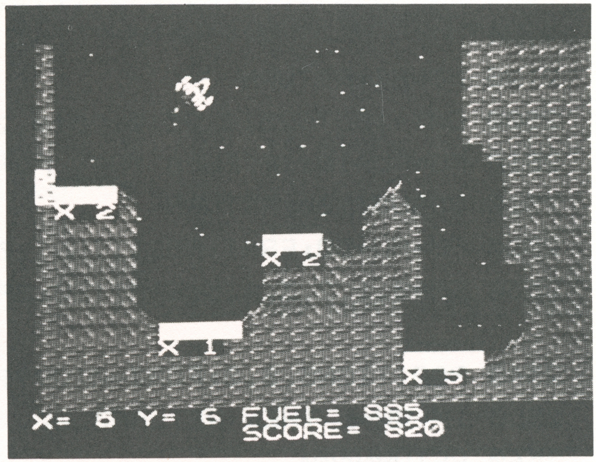
スプライト番号		定義している行
0	ニタニタ	2110行
1～3	ファイターフライ	2020行
4	スピナー	2050行
5	ファイアーボール	2080行

```
2080 DEFMOVE(I)=SPRITE(5,1,2,88,0,3):POSITIONI,CX,
200
2090 MOVEI:RETURN
2100 IF RND(NI)>0 THEN RETURN
2110 DEFMOVE(I)=SPRITE(11,7,2,64,0,3):POSITIONI,14
4,24
2120 MOVEI:RETURN
2500 ONI+1GOTO2510,2510,2510,2510,2520,2530
2510 FU=FU-10:IF FU<0 THEN FU=0
2515 GOSUB3200:RETURN
2520 FU=FU+50:ERAI:POSITIONI,0,0:GOSUB3200:RETURN
2530 ERA0,1,2,3,4,5:SPRITE0:SPRITE1,CX,120:SH=SH-1

2540 POSITION5,0,0:FORJ=0TO999:NEXT
2550 IF SH=0 THEN RETURN3500
2560 RETURN400
3000 LOCATE19,6:PRINTSC:RETURN
3100 LOCATE19,2:PRINTHS:RETURN
3200 LOCATE19,11:PRINTFU;" ":RETURN
3300 LOCATE20,19:PRINTSH;" ":RETURN
3500 LOCATE5,5:PRINT"GAME OVER"
3510 IF SC>HS THEN HS=SC:GOSUB3100
3520 GOTO300
```

アステロイド





宇宙空間でのファイターフライの攻撃もかわして，いよいよ小惑星が近づいてきた。だが，ファイターフライとの戦いで自動着陸装置をはじめ，いくつかのシステムが壊れてしまった。したがって小惑星への着陸はマニュアル（手動操縦）で行なわなければならない。しかも最初予定していた着陸地点までは，自動装置なしでは行きつけそうもない。手近な所で比較的安全な地点を探さなければ…。

よし，あの辺に降りよう。多少危険は残りそうだが，この際しかたがない。何カ所か平らな範囲もありそうだ。

しかし，燃料がもう残り少なくなっているから，進入のやり直しはできそうもない。こうなったら一発勝負に賭けるしかなさそうだ！

★ゲームの説明

昔，ゲームセンターにあった宇宙船の着陸ゲームのファミコン版です。

プログラムを走らせると

PUSH START BUTTON!!

の表示が出ますから，コントローラーのスタートボタンを押してください。

ゲームが始まると，画面に小惑星の地表の様子が出ます。中に4カ所白い部分があります

プログラムリスト （BG画面データーは96ページ）

```
10 CLS:CGSET 1,1:SPRITEON
110 DEFSPRITE0,(0,1,0,0,0)="ﾃﾞﾄﾞﾊﾞﾋﾞ"
120 DEFSPRITE1,(0,1,0,0,0)="ﾌﾞﾍﾞﾎﾞﾊﾟ"
130 DEFSPRITE2,(0,1,0,0,0)="ﾋﾟﾌﾟﾍﾟﾎﾟ"
140 DEFSPRITE3,(0,1,0,1,0)="ﾍﾞﾌﾞﾊﾟﾎﾞ"
150 DEFSPRITE4,(0,1,0,1,0)="ﾄﾞﾃﾞﾋﾞﾊﾞ"
155 DEFSPRITE5,(3,0,0,0,0)=CHR$(213)
156 DEFSPRITE6,(3,1,0,0,0)=CHR$(176)+CHR$(177)+CHR
$(178)+CHR$(179)
157 DEFSPRITE7,(3,1,0,0,0)=CHR$(180)+CHR$(181)+CHR
$(182)+CHR$(183)
160 DIMFX(4),FY(4),LS(4)
170 FORI=0TO4:READFX(I),FY(I),LS(I):NEXT
180 DATA15,4,209,12,12,211,4,16,213,-3,12,211,-6,4
,209
200 PALET B 0,47,48,48,48:VIEW
210 SC=0:Z=2:V=0:F=1000
220 LOCATE 5,5:PRINT"PUSH START BUTTON!!"
230 IF STRIG(0)<>1 GOTO 230
240 VIEW
500 HD=2:VX=V:VY=V:V=V+2:X=60*Z:Y=24*Z
510 FL=F:IFF>500THENF=F-50
550 LOCATE 10,22:PRINT"SCORE=";SC
1000 LOCATE0,21:PRINT"X=";VX;" Y=";VY;" FUEL=";FL;
" "
1010 K=STICK(0):IFFL=0ORK=0ORK>2THEN1050
1015 SPRITEHD
1020 IFK=1THENHD=HD+1:IFHD>4THENHD=4
```

が，これが着陸可能な範囲です。その下に，×１，×２，×５とあるのは，着陸地点の難易度によるボーナスです。

同時に，画面左上にスターシップが表示されますので，コントローラーでうまく姿勢制御を行ない，上手に着陸地点まで誘導してください。着陸地点以外の所に接触すると，スターシップは爆発してしまってゲームオーバーになってしまいます。

スターシップの誘導のために，姿勢制御とロケット噴射を使います。姿勢制御のためには，コントローラーの十字ボタンを使い，スターシップの向きを変えます。一回の姿勢変更は燃料を５ユニット消費します。また，コントローラーのＡボタンを押すとロケットが噴射され，これはスピードの調整に使います。一回の噴射で燃料を10ユニット消費します。

速度は画面の左下に表示されます。Ｘが横方向のスピードで，右向きが＋です。Ｙは垂直方向のスピードで，下向きが＋です。何もしないと，１単位時間にＹ方向のスピードが＋１されます。ロケットを噴射すると，１単位時間あたりＹ方向のスピードが－２されます。スターシップが斜めの時にロケットを使うと，Ｘ，Ｙ方向共にスピードが１ずつ変化します。

着陸のための条件は，Ｘ方向のスピードの絶対値が２以下，Ｙ方向が５以下であり，かつスターシップがまっすぐ上を向いていることで，これ以外の場合は爆発してしまいます。

着陸に成功すると，着陸点の倍数に，残りの燃料をかけただけの得点が加算され，スタートボタンを押すと次のパターンに進みます。１面進むごとに，燃料から50ユニットずつ減らされます。また，スターシップが最初からＸ，Ｙ方向にスピードを持った状態から始まり，だんだん困難になっていきます。

★プログラムの説明

●初期設定　(100～550)

スプライトの定義，パレットのセット，変数の初期化，画面の設定を行ないます。

●姿勢制御　(1000～1050)

キー入力をチェックして，スターシップの姿勢の変更をします。スターシップの向きは変数HDにセットされています。

●スピード変更　(1060～1140)

ボタンＡが押されていれば，スターシップの向きに応じたスピードの変更を行ないます。

●座標チェック　(1150～1300)

新しい座標を計算して，その座標のBG面に書かれている物を調べ，コードをＣにセットします。Ｃの各種類に応じた処理に飛びます。

●着陸処理　(2000～2040)

着陸のための条件をチェックして，満たしていれば着陸成功です。残り燃料を得点に加算

```
1030 IFK=2THENHD=HD-1:IFHD<0THENHD=0
1040 FL=FL-5
1045 DEFSPRITE5,(3,0,0,-(HD=3),0)=CHR$(LS(HD))
1050 SPRITEHD,X/Z,Y/Z
1060 S=STRIG(0):VY=VY+1
1070 IFS<>8THEN1150
1080 SPRITE5,X/Z+FX(HD),Y/Z+FY(HD):FL=FL-10
1090 ONHD+1GOTO1100,1110,1120,1130,1140
1100 VX=VX-3:GOTO1150
1110 VX=VX-1:VY=VY-1:GOTO 1150
1120 VY=VY-3:GOTO 1150
1130 VX=VX+1:VY=VY-1:GOTO 1150
1140 VX=VX+3
1150 X=X+VX:Y=Y+VY:IF Y/Z<24THENY=24*Z
1160 BX=(X/Z-16)/8:BY=(Y/Z-24)/8
1170 C=ASC(SCR$(BX,BY))
1180 IF NOT(C=32 OR C=205 OR C=206) THEN 2500
1190 C=ASC(SCR$(BX+1,BY))
1200 IF NOT(C=32 OR C=205 OR C=206) THEN 2500
1210 C=ASC(SCR$(BX,BY+1))
1230 IF NOT(C=32 OR C=205 OR C=206) THEN 2500
1240 C=ASC(SCR$(BX+1,BY+1))
1260 IFNOT(C=32ORC=205ORC=206)THEN2500
1270 C=ASC(SCR$(BX,BY+2))
1280 IF C>252THEN2000
1290 C=ASC(SCR$(BX+1,BY+2))
1300 IFC>252THEN2000
1310 SPRITE5:GOTO1000
2000 IFABS(VX)>2ORABS(VY)>5 ORHD<>2THEN2500
2010 SC=SC-((C=253)+(C=254)*2+(C=255)*5)*FL
```

します。

●**ゲームオーバー　(2500〜2540)**

スプライトを消します。

★プログラムの改造

燃料を増やしたい人は210行のF＝1000を変えてください。同じく210行のZ＝2を適当に大きくしてやるとスターシップは反応が鈍くなります。

地形を変えたい人は，自分でBG画面を作ってください。ただしこの時，着陸地点は次のキャラクターを使います。

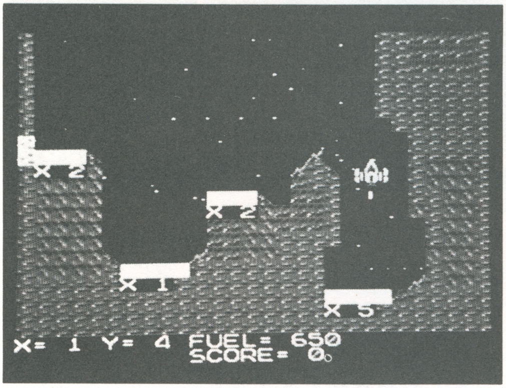
×1⟶M5，×2⟶M6，×5⟶M7

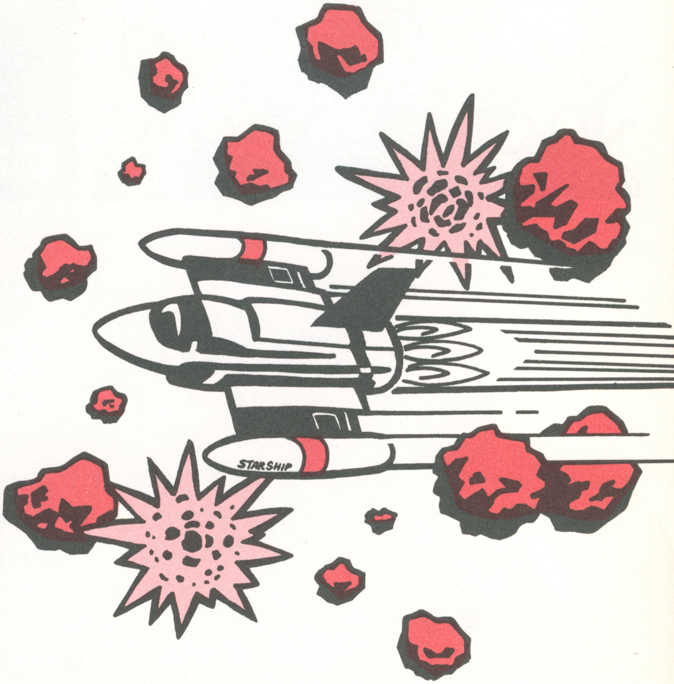
★変数表

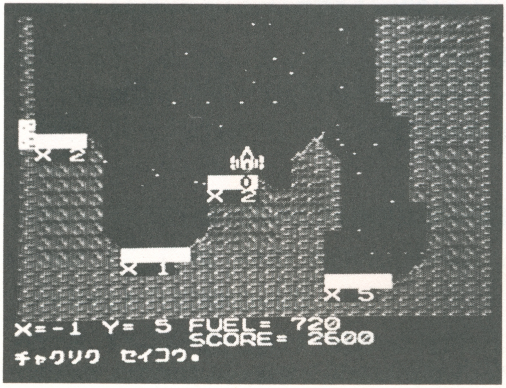
X，Y	スターシップ座標
VX，VY	スターシップのスピード
HD	スターシップの向き
FL	燃料
V	スターシップの初期速度
SC	得点



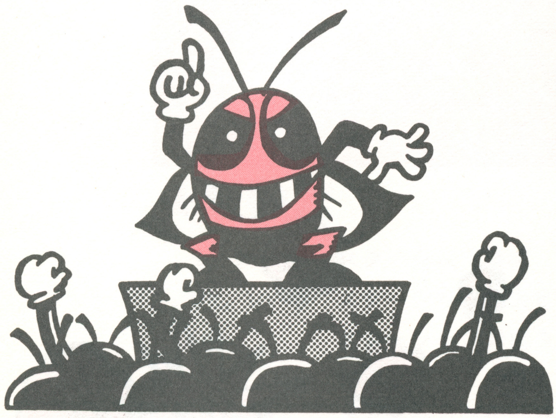
```
2020 LOCATE0,23:PRINT"ﾁｬｸﾘｸ ｾｲｺｳ｡";
2030 IFSTRIG(0)<>1THEN2030
2040 LOCATE0,23:PRINT"                    ";:GOTO 500
2500 SPRITEHD:SPRITE5
2510 FORI=0TO4:SPRITE6,X/Z,Y/Z:FORJ=0TO799:NEXT
2520 SPRITE6:SPRITE7,X/Z,Y/Z:FORJ=0TO500:NEXT:SPRI
TE7:NEXT
2530 LOCATE10,8:PRINT"GAME OVER"
2540 GOTO 210
```

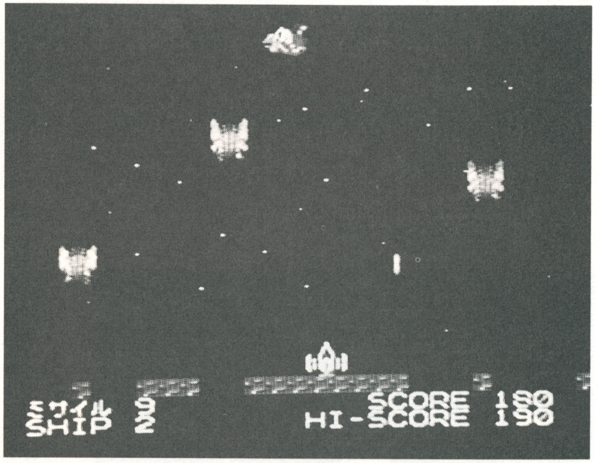






FLYFALL





スターシップは，無事に小惑星“S3612-09”に着陸できた。だが，着陸地点は，調査隊の位置から少し離れている。さっそく救援に向かおうとした時，着陸地点の上空に黒い点がポツポツと現われた。まさか…，やっぱりファイターフライだ。何てしつこいハエ達なんだ!!

しかも奴らめ，我々が無事着陸したのを見て，捨て身で攻撃して来た。何と，奴らが落ちた所は，地面に大穴があいてしまうではないか。あんなのに体当りされたら，いくらなんでもたまらない。落ちる前に撃たなくては…

しかし攻撃用にエネルギーを使うわけにはいかないので，古典的なタマで攻撃しなくてはならない。タマには限りがあるが，さいわい敵か味方かよくわからないが，カラスのような奴が補給用のタマを落としていく。あれを拾いながら戦うしかなさそうだ。やれやれ……。

★ゲームの説明

次々と降ってくるファイターフライを撃ち落としてください。スターシップはコントローラーⅠで動かします。

プログラムを走らせると，例によって，

PUSH START BUTTON!

が表示されますから，コントローラーⅠのスタートボタンを押してゲームを始めます。

プログラムリスト　（BG画面データーは97ページ）

```
100 CLS:CGEN2:CGSET1:SPRITEON
110 PALET S 1,47,48,21,18
120 PALET S 2,48,48,38,18
130 PALET S 3,47,48,48,48
140 DEFSPRITE0,(0,1,0,0,0)="ﾋﾟﾌﾟﾍﾟﾎﾟ"
150 DEFSPRITE1,(0,1,0,0,0)=CHR$(180)+CHR$(181)+CHR
$(182)+CHR$(183)
200 LOCATE5,8:PRINT"PUSH START BUTTON!"
210 IF STRIG(0)<>1 GOTO210
220 SC=0:SH=3
300 M=10:CX=14:AX=-1:C=0
302 VIEW:GOSUB6000:GOSUB6100:GOSUB6200:GOSUB6300
310 SPRITE0,CX*8+16,168:SPRITE1
320 POSITION5,0,0
500 IFSTRIG(0)=8GOSUB1000
510 S=STICK(0):X=CX-(S=1)+(S=2):IFS=0GOTO550
520 IFX<0THENX=0
530 IFX>26THENX=26
540 IFSCR$(X-(S=1),20)<>" "THENCX=X:SPRITE0,CX*8+1
6,168
550 IFC>0AND(CX=AX OR CX+1=AX)GOSUB2000
555 IFC>0THENC=C-1:IFC=0THENLOCATEAX,19:PRINT" ":A
X=-1
556 IFYPOS(5)=176GOSUB4500
560 IF C=0 AND MOVE(5)=0 AND MOVE(4) AND XPOS(4)<A
X*8+16THENMOVE5
570 X=CX*8+16:FORS=0TO3:I=S
580 IFABS(XPOS(I)-X)<16ANDYPOS(I)>152THENS=4:NEXT:
GOTO3000
```

スターシップは，コントローラーの十字ボタンで左右に動かします。この時，動けるのは地面の上だけです。また，ファイターフライが落ちてできた穴は越えられません。

タマの発射は，コントローラーのＡボタンです。最初は，タマが10発しかありません。残りのタマ数は画面の左下に表示されていますから，慎重に狙ってください。

10発撃ったら終わりではあんまりですから，タマの補給があります。画面の上の方をカラス（ニットピッカー）が飛んでいく途中で，タマを投下していきます。このタマは地上について，そこに地面があれば“@”が表示されます。これをスターシップで回収（そこへ行くだけでよい）すると，タマが10発増えます。ただしスターシップは20発までしかタマを持てませんから，オーバーした分は捨てられてしまいます。@はしばらくすると消えてしまいますから，素早く拾ってください。

ファイターフライは１匹10点で，500点ごとにスターシップが１機増えます。

★プログラムの説明

●初期設定　（100～320）

スターシップと爆発用のスプライトの定義，パレットの色セット，変数の初期セットなどを行ない，画面の表示を行ないます。

●スターシップ移動　（500～540）

キー入力をチェックして，Ａボタンが押されていればタマ発射の処理（1000行～）へ，十字ボタンなら左右の移動を行ないます。

●ファイターフライ処理　（570～620）

スターシップの位置（スプライト座標）を変数Ｘにセットして，ファイターフライとの位置を比較して，衝突しているかを調べます。衝突していればその処理（3000行）に飛びます。同様にしてタマがファイターフライに当ったかを調べて，当っていれば3500行からの処理を呼びます。ファイターフライが地面に着いていたら穴掘り処理(4000行)，表示されていなければ発生処理（5000行）を呼びます。以上を４回（４匹）くり返します。

●補給発生処理　（650～670）

すでに発生していなければ，1/50の確率でタマの補給が生じます。タマの投下位置は乱数で決め，変数AXにセットしておきます。

●タマ発射処理　（1000～1010）

タマを発射済みか，残りのタマがない場合は何もしないで戻ります。そうでなければ，タマを撃ちます。

●補給獲得　（2000～2010）

残りのタマ数を10発増やします。20発を越えた分は加算されません。

```
585 IF NOT MOVE(6) GOTO600
590 IF XPOS(I)-6<XPOS(6) AND XPOS(I)+14>XPOS(6) AN
D YPOS(I)+14>YPOS(6) AND YPOS(I)-6<YPOS(6) GOSUB35
00
600 IF YPOS(I)=168GOSUB4000
610 IF NOT MOVE(I)GOSUB5000
620 NEXT
630 IF NOT MOVE(6)THENERA6:POSITION6,255,255
640 IF XPOS(4)=16THENERA4:POSITION4,255,0
650 IF AX>0 OR RND(50)>0GOTO500
660 DEFMOVE(4)=SPRITE(15,7,2,108,0,2)
662 AX=RND(28):POSITION4,232,24:MOVE4
665 DEFMOVE(5)=SPRITE(12,5,1,68,0,3)
670 POSITION5,AX*8+16,40:GOTO500
1000 IFMOVE(6)ORM=0THENRETURN
1005 DEFMOVE(6)=SPRITE(12,1,1,64,0,0)
1010 POSITION6,CX*8+20,160:MOVE6:M=M-1:GOSUB6000:R
ETURN
2000 M=M+10:IFM>20THENM=20
2005 LOCATEAX,19:PRINT" "
2010 GOSUB6000:AX=-1:C=0:RETURN
3000 ERAI:SPRITE0:SPRITE1,CX*8+16,168:SH=SH-1:POSI
TIONI,0,0
3010 FORJ=0TO999:NEXT:ERA0,1,2,3,4,5,6
3020 IFSH=0GOTO10000
3030 GOTO300
3500 ERAI,6:POSITIONI,0,0:POSITION6,255,255:SC=SC+
10:GOSUB6100
3510 IF(SC MOD500)=0THENSH=SH+1:GOSUB6300
3520 RETURN
```

●ファイターフライとスターシップの衝突　(3000〜3030)

爆発の表示を行ない，スターシップを１機減らします。残りが０ならゲームオーバーです。

●タマ命中　(3500〜3520)

得点に10点加えます。500点につきスターシップ１機がボーナスとして増えます。

●穴掘り処理　(4000〜4020)

ファイターフライが落ちた所に，穴をあけます。そこに補給のタマがあった場合は，それはクリアされてしまいます。

●ファイターフライ発生　(5000〜5010)

確率1/50でファイターフライを発生させます。出現位置は乱数で決定します。

●データ表示用ルーチン　(6000〜6300)

残りタマ数，得点，ハイスコア，残りスターシップ数を表示します。

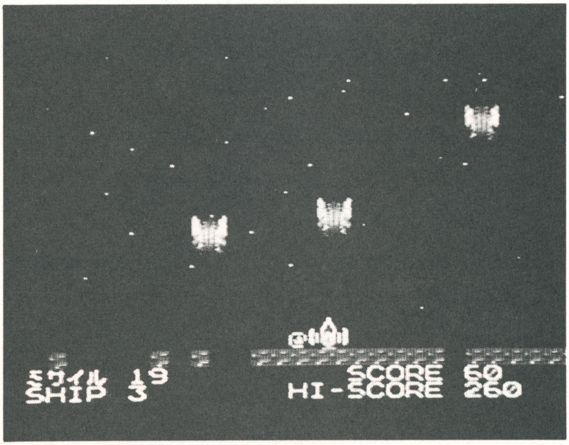
●ゲームオーバー　(10000〜10020)

ゲームオーバーのための処理です。

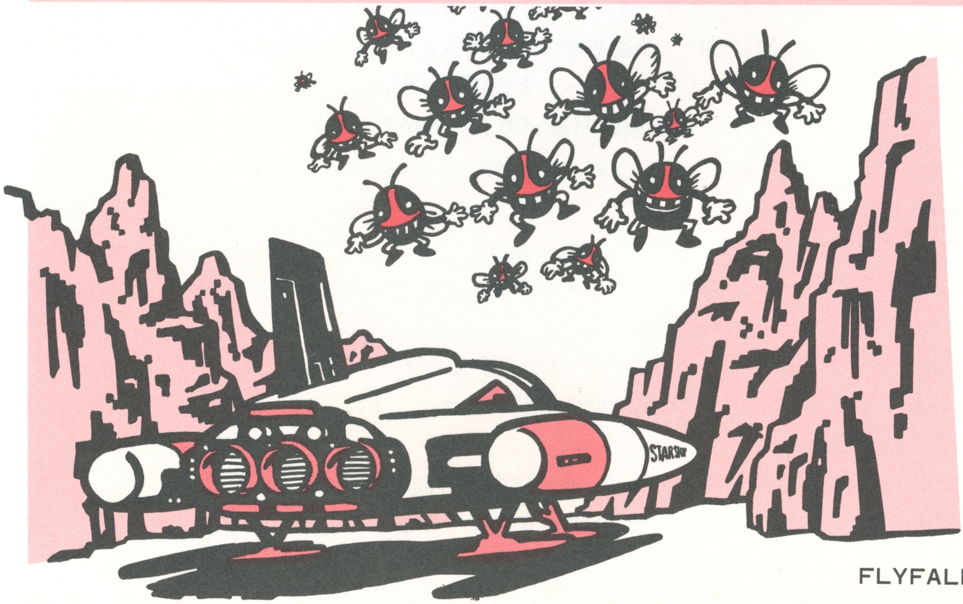
★変数表

CX	スターシップ位置
AX	補給投下位置
M	残りタマ数
C	補給消滅までの時間
SH	スターシップ残り機数
SC	得点
HS	ハイスコア

スプライト (DEF MOVE)	
0〜3	ファイターフライ
4	ニットピッカー
5	補給用タマ
6	タマ

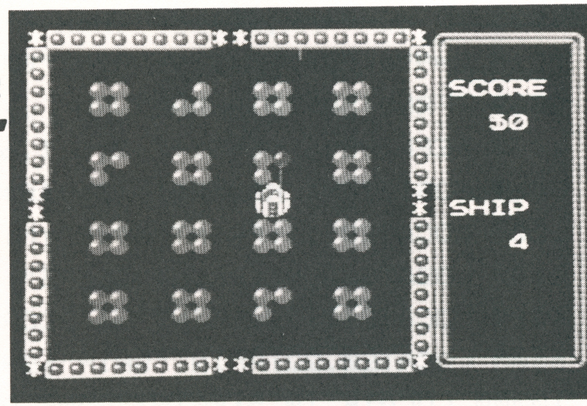


```
4000 J=(XPOS(I)-16)/8:LOCATEJ,20:PRINT" ":ERAI
4010 IF C=0 OR J<>AX GOTO4020
4015 LOCATEJ,19:PRINT" ":C=0:AX=-1
4020 POSITIONI,0,0:RETURN
4500 POSITION5,0,0:ERA5
4505 IF SCR$(AX,20)=" "THEN AX=-1:RETURN
4510 C=99:LOCATEAX,19:PRINT"@":RETURN
5000 IFRND(50)>0THENRETURN
5005 DEFMOVE(I)=SPRITE(2,5,1,60,0,1)
5010 POSITIONI,RND(211)+16,48:MOVEI:RETURN
6000 LOCATE0,21:PRINT"ﾐｻｲﾙ";M;" ":RETURN
6100 LOCATE16,21:PRINT"SCORE";SC:RETURN
6200 LOCATE13,22:PRINT"HI-SCORE";HS:RETURN
6300 LOCATE0,22:PRINT"SHIP";SH;" ":RETURN
10000 LOCATE 10,5:PRINT"GAME OVER"
10010 IF SC<=HS GOTO 200
10020 HS=SC:GOSUB6200:GOTO200
```



コマンダー





次々と降ってくるファイターフライを，何とか撃退してほっとしたのもつかの間，今度は敵の地上部隊に包囲されてしまった。包囲を打ち破らなければ味方は全滅だ。現在使用可能なのは１人乗りの小型戦車「スターキラー」が数台あるだけだ。これで，増援が到着するまで敵と戦かわなくてはならない。前進基地の運命は，君の活躍にかかっている!!

コントローラーを使って「スターキラー」を操作し，周囲をとりかこむ敵の砲を破壊してください。「スターキラー」の方が射程が長いのと，戦場にある障害物をうまく利用しないと，すぐやられてしまいます。

★遊び方

ゲームが始まると，画面の中央に「スターキラー」が表示されます。周囲の“◘”が敵の砲，赤いボールみたいなのは障害物です。

「スターキラー」の操作は，コントローラーＩを使います。移動は十字ボタンで行ないます。移動方向は上下左右だけで，ななめには動けません。また，前方に障害物や，敵の砲がある場合も動けません。

A, Bの各ボタンでタマを発射します。タマは進行方向に発射され，Aボタンを押すと，「スターキラー」の中心線の右側から，Bボタンを押すと左側から発射されます。タマは，一度

プログラムリスト　（BG画面データーは98ページ）

```
100 CLS:CGSET1,1:SPRITEON:DIM DX(7),DY(7)
120 FORI=0TO7:READDX(I),DY(I):NEXT:DATA 0,-1,1,-1,
2,0,2,1,1,2,0,2,-1,1,-1,0
140 DEFSPRITE0,(3,1,0,0,0)=CHR$(180)+CHR$(181)+CHR
$(182)+CHR$(182)
250 N=0:R=40:SH=4:B=1000
300 VIEW:E=64
400 FORI=0TO6:ERAI:POSITIONI,0,0:NEXT:SPRITE0:CX=9
:CY=9:D=1:GOSUB3000:GOSUB3100
420 DEFMOVE(7)=SPRITE(8,1,4,4,0,1):POSITION7,88,10
4:MOVE7
500 S=STRIG(0):IFS>0GOSUB1000
510 S=STICK(0):IFS=0ORMOVE(7)GOTO560
520 S=-(S=1)*3-(S=2)*7-(S=4)*5-(S=8):I=(S-1)/1:F=0
:IFS=0GOTO560
530 IFSCR$(CX+DX(I),CY+DY(I))=" "ANDSCR$(CX+DX(I+1
),CY+DY(I+1))=" "THENF=-1
540 IFS<>D THENDEFMOVE(7)=SPRITE(8,S,2,4,0,1):D=S
545 POSITION7,CX*8+16,CY*8+24:IFF THENCX=CX-(S=3)+
(S=7):CY=CY-(S=5)+(S=1)
550 MOVE7:IFNOT F THENFORI=0TO88:NEXT:CUT7
560 IFNOT MOVE(6)GOTO640
570 X=(XPOS(6)-16)/8:Y=(YPOS(6)-24)/8
571 IFX<1ERA6:X=0
572 IFX>18ERA6:X=19
573 IFY<1ERA6:Y=0
574 IFY>18ERA6:Y=19
580 I=ASC(SCR$(X,Y))
```

に一発ずつしか撃てません。

敵のタマには，障害物をすり抜けてくるものもあるので，障害物の後でもノンビリしているとやられてしまいます。

また，敵の砲は全部で64個で，破壊されると，“＊”になります。64個の砲をすべて破壊すると一面クリアで，次の面に移ります。面が進むにつれて敵のタマの数が増え，射程も長くなって難しくなってきます。がんばってください。まあ，最初のうちは一面クリアするのも大変だと思いますが……。

敵の砲を破壊すると20点，障害物は５点です。1000点ごとに「スターキラー」が１台増えます。

★プログラムについて

プログラムについては，ファミリーベーシック Ver.2 のフリーエリアをほぼ限界まで使用しています。したがって，プログラムを入力する際には，必ずリストのとおりに入力してください。勝手にプログラム中にスペースをいれたりするだけでメモリ不足（OM エラー）になる可能性があります。ただし，リスト中に入っているスペースはそのとおりに入れてください。取ってしまうと正しく動いてくれません。あと，IF～THEN 文の THEN が抜けている所がありますが，これは間違いではありません。

プログラムは，大きく分けて次の様になります。

100～　イニシャライズ
400～　画面設定。座標セット
500～　メインルーチン
1000～　スターシップ　タマ発射
1500～　敵　タマ発射
2000～　スターシップ被弾
2500～　一面クリア
3000～　表示用サブルーチン

このうち，メインルーチン（500～）と敵のタマ発射ルーチン（1500～）を簡単に説明しておきます。まず，メインルーチンは，キー入力と「スターキラー」移動（500～550行），タの位置チェック（560～630行），敵のタマチェック（640～690行）に分かれています。

タマのチェック（タマが障害物に当たったかどうか）はタマの位置の BG　GRAPHIC 座標を計算して，SCR$関数を使って直接 BG 画面に何が描かれているかを調べ，「スターキラー」に当たったかどうかは，両方のスプライト座標をそのまま比較しています。

敵のタマ発射ルーチンで重要なのは，射撃の正確さの決め方で，あまり正確に撃つように

```
590 IFI=32GOTO640
600 IFI=207THENSC=SC+5:LOCATEX,Y:PRINT" ":GOTO620
605 IFI<>28GOTO640
610 LOCATEX,Y:PRINT"*":SC=SC+20:E=E-1:IFE=0GOTO250
0
620 ERA6:GOSUB3000
640 FORI=0TON:X=(XPOS(I)-16)/8:Y=(YPOS(I)-24)/8
650 IFX=-2GOSUB1500
660 IFNOT MOVE(I)THENERAI:POSITIONI,0,0:GOTO690
670 IFXPOS(7)-6<XPOS(I)ANDXPOS(7)+14>XPOS(I) AND Y
POS(7)-8<YPOS(I)ANDYPOS(7)+16>YPOS(I)GOTO2000
680 IFASC(SCR$(X,Y))=207THENLOCATEX,Y:PRINT" ":POS
ITIONI,0,0:ERAI
690 NEXT
700 GOTO500
1000 IFMOVE(6)RETURN
1010 DEFMOVE(6)=SPRITE(12,D,1,80,0,3):X=XPOS(7):Y=
YPOS(7)
1020 IFS=8THENON D/2+1GOTO1050,1060,1070,1040
1030 ON D/2+1GOTO1040,1050,1060,1070
1040 POSITION6,X,Y:GOTO1080
1050 POSITION6,X+8,Y:GOTO1080
1060 POSITION6,X+8,Y+8:GOTO1080
1070 POSITION6,X,Y+8
1080 MOVE6:RETURN
1500 X=0:Y=0:IFRND(2)=0GOTO1530
1510 IFRND(2)=0THENX=19
1520 Y=(CY+16+RND(8))MOD20:GOTO1550
1530 IFRND(2)=0THENY=19
1540 X=(CX+16+RND(8))MOD20
```

してしまうとゲームが難しく，というよりはつまらなくなってしまいます。そこで，このゲームでは「スターキラー」の位置を中心にして±座標ぐらいの範囲でタマの発射位置を決めています。

実際には，1520行と1540行で乱数を使って，多少ズレるようにしてあります。今は，

〜＋16＋RND (8) 〜

となっていますが，もっと正確に（難しく）したい人は，この8を減らして，減らした数の半分を16に加えてください（たとえば　〜＋17＋RND（6）〜にする)。もっとデタラメにしたい人は，逆にすればいいわけです。

他に可能な改造は，

●敵の射程を変える

250行のR＝40を修整（小さくすると短かくなる）

●敵のタマの最大数を減らす

2500行のN＝(N＋1) MOD 4の“4”を1〜4の範囲で変える。この式自体をなくしてしまうと，タマは常に1発で増えない。

●スターキラーの数を変える

250行のSH＝4を，好きな数値に変える。

などがあります。これ以外の変更は大変ですが，参考として変数表を載せておきます。あと，画面の障害物の位置はBG面のデータを変えれば，好きなように変更できます。ただし，ゲーム開始時にスタートキラーが現われる位置には，障害物を置かないでください。

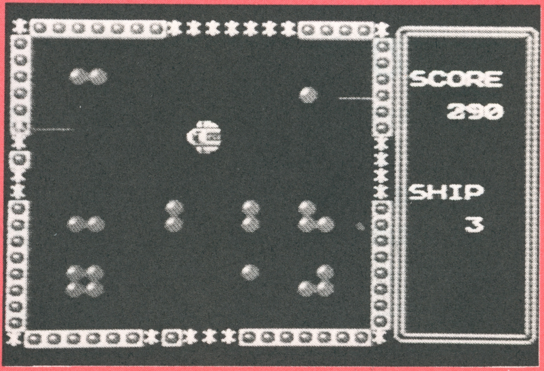
★変数表

D	スターキラーの向き
CX，CY	スターキラーの現在位置（BG座標）
R	敵の射程
N	敵のタマ数
SC	得点
X，Y	各種座標用
E	敵の砲の数

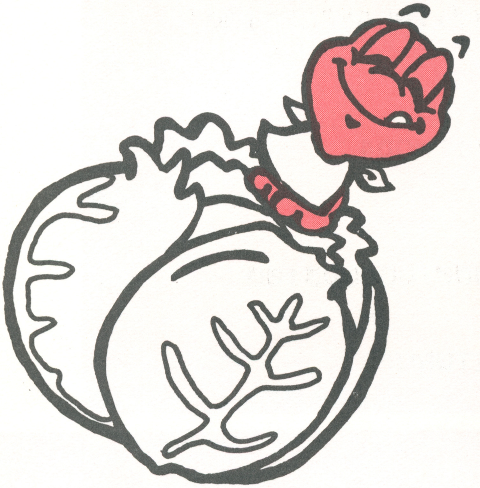
スプライト（DEF MOVE）	
0〜5	敵のタマ
6	味方のタマ
7	スターキラー

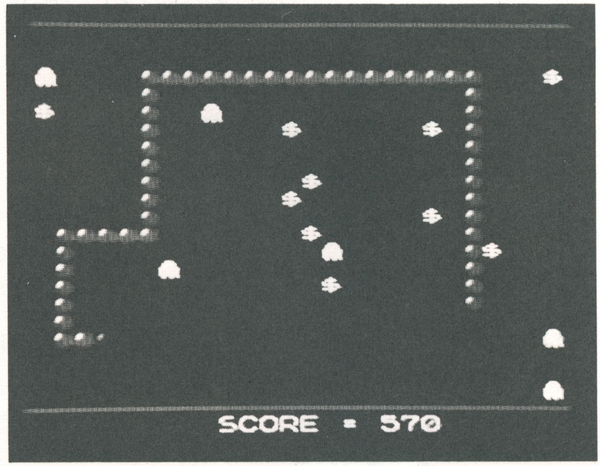
```
1550 IFSCR$(X,Y)="*"RETURN
1560 H=-(Y=19)-(X=0)*3-(Y=0)*5-(X=19)*7
1570 DEFMOVE(I)=SPRITE(12,H,1,R,0,3):POSITIONI,X*8
+16,Y*8+24:MOVEI:RETURN
2000 ERAI:SPRITE0,XPOS(7),YPOS(7):PLAY"C2E1G0"
2010 SH=SH-1:IFSH<0END
2020 GOTO400
2500 R=R+8:IFR>64THENR=40:N=(N+1)MOD4
2510 PLAY"C1GDEF"GOTO300
3000 LOCATE22,5:PRINTSC
3010 IFSC>=B THENSH=SH+1:B=B+1000:GOSUB3100
3020 RETURN
3100 LOCATE23,12:PRINTSH;" ":RETURN
```





いも虫ごろごろ





このゲームは，コンピューター・ゲームの大古典ともいえます。おそらく，ほとんどのコンピューター用に発表，発売されているでしょう。

ルールは簡単です。丸い球をコントローラーで上下左右に動かしてください。「$」がエサです。「$」を食べると，いも虫の長さが一つふえます。かべにぶつかったり，自分の体にぶつかるとGAME OVERです。またレベル8～9では，ある長さ以上になると，「ᗣ」が出てきます。この「ᗣ」にぶつかっても，GAME OVERです。「$」1個食べるごとに得点は10点プラスされます。

★遊び方

「RUN」すると，ゲームレベルを聞いていきます。0～9の数字キーを押してください。これは，いも虫の移動スピードを変えます。0が一番遅く，9が一番速くなっていきます。この時，RETURNキーは押す必要がありません。

得点は画面下に表示されます。GAME OVERになるとその回までのハイスコアが表示されます。ハイスコアは，ストップキーでゲームをとめたり，電源を切らない限り残ります。

このゲームにはいろんなバリエーションがあります。たとえば，砂漠でエサさがし。この場合，エサを1つ食べるとエネルギーが20にふえます。そして1移動ごとにエネルギーが1

プログラムリスト

```
10 DIM A(1,50)
20 FOR I=0 TO 50:A(0,I)=0:A(1,I)=0:NEXT
40 CLS:LOCATE 0,10:PRINT "LEVEL LOW=0  ---  HIGH=9
"
45 A$=INKEY$:IF A$<"0" OR A$>"9" THEN 45
50 WA=270-VAL(A$)*30
60 CLS
70 LOCATE 0,0:PRINT CHR$(239);:FOR I=1 TO 26:PRINT
 CHR$(237);:NEXT:PRINT CHR$(240);
80 FOR I=1 TO 21:LOCATE 0,I:PRINT CHR$(238);:LOCAT
E 27,I:PRINT CHR$(238);:NEXT
90 LOCATE 0,22:PRINT CHR$(241);:FOR I=1 TO 26:PRIN
T CHR$(237);:NEXT:PRINT CHR$(242);
110 FOR I=0 TO 9
120 LOCATE RND(26)+1,RND(21)+1:PRINT "$";
130 NEXT
140 AX=14:AY=2:AH=4:AL=1:AF=0:SC=0
150 A(0,0)=14:A(1,0)=2
200 GOSUB 900
210 K=STICK(0):IF K<>0 THEN AH=K
220 FOR I=0 TO WA:NEXT
230 GOTO 200
900 IF AH=4 THEN AY=AY+1
910 IF AH=2 THEN AX=AX-1
920 IF AH=8 THEN AY=AY-1
930 IF AH=1 THEN AX=AX+1
1000 AF=AF+1:IF AF=51 THEN AF=0
1010 AE=AF-AL:IF AE<0 THEN AE=AE+51
```

マイナスされ，0になるまでエサを食べないとエネルギー切れになります。これも，もうひとひねりして，エサを一つたべれば長さが一つふえ，エネルギーを10消費するごとに長さを一つへらす。というのも考えられます。
たとえば，140行と1350行を変更し，1005行を追加します(**リスト-2**)。これで，いも虫は20回移動をくり返すまでにエサをとらなければGAME OVERになってしまいます。

このプログラムの特長は，いも虫の長さが長くなっても移動の速度を変えなくするため，常に頭としっぽの部分しか見ていない，という点です。いも虫の各部分の座標は配列変数A( )の中に入ってますが，先頭＝AF，と，長さ＝ALだけですべてを管理しています。そして配列変数A( )を使いきるとまたはじめから登録しています。したがってヘビの長さを変える時は，10行，1000行，1010行，1350行を変えてください。短くする時は1350行だけでいいのですが……。

★プログラムの説明

10	配列宣言
20	配列変数の初期化
40～50	レベル選択
60～90	わくを書く
110～130	最初のエサの配置
140～150	変数の初期化
200	いも虫移動のサブルーチンコール
210	方向変換のよみとり
220	レベルに応じた時間待ち
230	200行へもどる
900～930	進行方向によるいも虫の先頭位置決定
1000～1010	ポインター変更
1020～1030	進行方向のキャラクターをしらべる
1100～1300	ポインターのしめす所にいも虫の先頭座標を登録，いも虫を表示，一番うしろの位置を消して，メインへ
1350	エサをたべた時の得点をプラスし，エサを消し，長さをふやす。長さが50以上なら障害物を置くサブルーチンを呼ぶ
1360～1390	減った分のエサを他へ置く
1400～1420	障害物をおく，サブルーチン
1500～1600	ハイスコアをきめて表示　キーを押すと20行へ

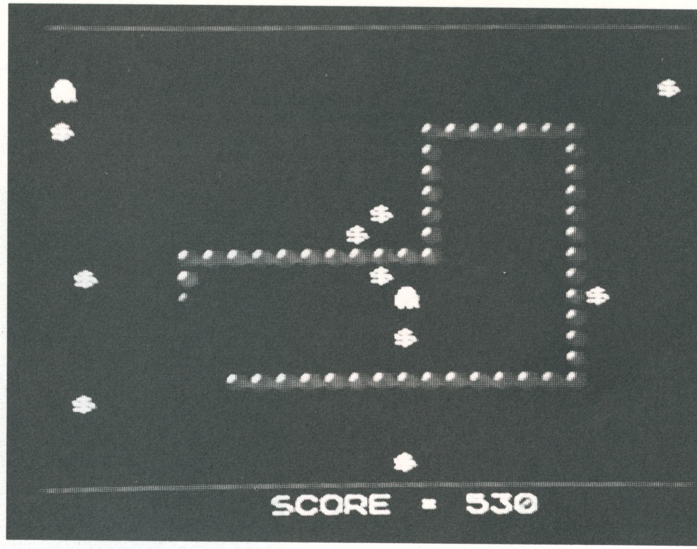
```
1020 IF SCR$(AX,AY)="$" THEN GOSUB 1350:GOTO 1100
1030 IF SCR$(AX,AY)<>" " THEN 1500
1100 A(0,AF)=AX:A(1,AF)=AY
1200 LOCATE AX,AY:PRINT CHR$(207);
1210 LOCATE A(0,AE),A(1,AE):PRINT " ";
1300 RETURN
1350 AL=AL+1:SC=SC+1:LOCATE 6,23:PRINT "SCORE =";S
C;"0  ";:LOCATE AX,AY:PRINT CHR$(207);:IF AL>50 TH
EN AL=50:IF WA<40 THEN GOSUB 1400
1360 BX=RND(26)+1:BY=RND(21)+1:IF SCR$(BX,BY)<>" "
 THEN 1360
1370 LOCATE BX,BY:PRINT "$";
1390 RETURN
1400 IF RND(5)*2-3<WA/10 THEN RETURN
1410 CX=RND(26)+1:CY=RND(22)+1:IF SCR$(CX,CY)<>" "
 THEN 1410
1420 LOCATE CX,CY:PRINT CHR$(219);:RETURN
1500 LOCATE 7,8:PRINT "  GAME OVER   ";
1510 IF SC>HS THEN HS=SC
1520 LOCATE 5,10:PRINT " HI-SCORE =";HS;"0  ";
1530 LOCATE 7,12:PRINT "  HIT START   ";
1540 IF STRIG(0)=0 THEN 1540
1600 GOTO 20
```

★変数表

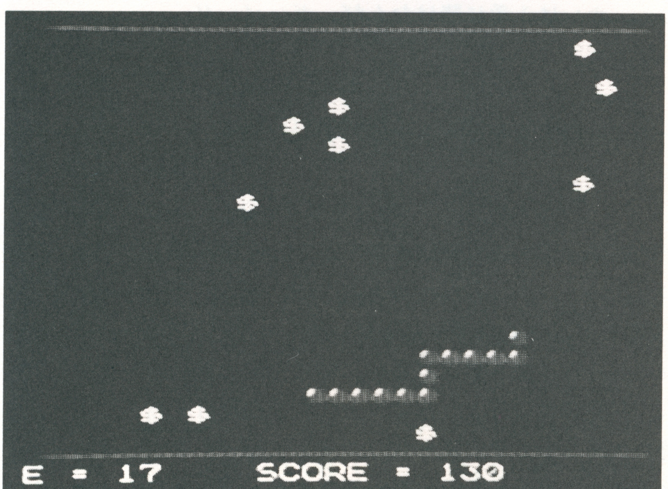
AX	いも虫の先頭のＸ座標
AY	いも虫の先頭のＹ座標
A(0, n) A(1, n)	順番にAX，AYを入れていく
AH	いも虫の進行方向，STICKに対応
AL	いも虫の長さ
AF	現在いも虫の先頭のX，Y座標が配列変数A(　)のどこに入っているかを示すポインター
AE	いも虫の一番うしろ（消さなければいけない場所）の座標A(　)のどこに入っているかのポインター
SC	得点
AX AY BX BY CX CY	ダミーのX・Y座標
WA	待ち時間（レベル）

プログラムリスト２

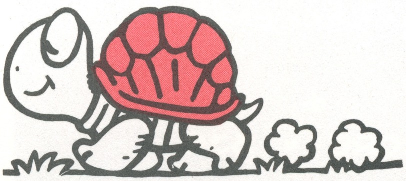
```
140 AX=14:AY=2:AH=4:AL=1:AF=0:SC=0:T=20
1005 T=T-1:LOCATE 0,23:PRINT "E =";T;"  ";:IF T<0
THEN 1500
1350 AL=AL+1:T=20:SC=SC+1:LOCATE 10,23:PRINT "SCOR
E =";SC;"0  ";:LOCATE AX,AY:PRINT CHR$(207);:IF AL
>50 THEN AL=50:IF WA<40 THEN GOSUB 1400
```

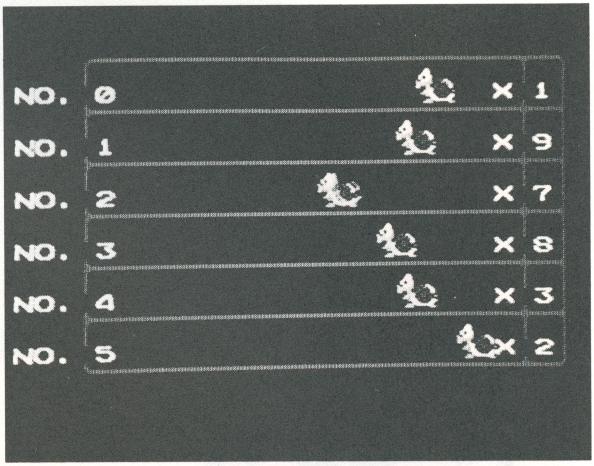

SCORE = 530




E = 17
SCORE = 130

カメレース





ほら，ガンバレ。止まってないで前に進んでくれ!!　こら，4番のカメさん。おまえに残りの金全部賭けたんだから，当たれば9倍の大穴だ。あっ，こらこら，1番と3番，先に行くんじゃない。

いや1番そのまま行ってしまえ。オレは1番に賭けたんだ。ガンバレ!!

んじゃ，こっちはどうなるんだ。

んなの知るか!!

以上，ある日のカメさんレース，白熱するギャラリーの声でした。

★ゲームの説明

全部で6匹のカメさんのレースです。1～4人で遊べます。1人で遊んでもつまらないので，できるだけ大勢で遊びましょう。

ゲーム開始時は各自500円（少ないと思う人はドルだと思いましょう）ずつ持っています。賭け方は単勝のみで，1着のカメさんを予想してください。賭け金の上限はありません（もちろん所持金の範囲です）から，一発勝負の好きな人は，大穴に全部賭けてみましょう。

全員が賭け終わったら，一勢にスタートです。あとはひたすら大騒ぎしながら見守ってください。競馬と違って，相手はカメさんですからなかなか進みません。完全に止まってしま

プログラムリスト　（BG画面データーは99ページ）

```
100 CGSET1,2:SPRITEON
110 DIM OD(5),MN(3),U(1,3),NA$(3),GL(5)
120 FOR I=0 TO 5
130 DEFMOVE(I)=SPRITE(13,7,3,4,0,3)
140 NEXT
150 CLS:INPUT"ﾅﾝﾆﾝ ﾃﾞ ｱｿﾋﾞﾏｽｶ (1-4)";PL
160 IF PL<1 OR PL>4 GOTO 150
170 PL=PL-1
180 FOR I=0 TO PL
190 PRINT "NO.";I;" ｻﾝ ﾉ ﾅﾏｴﾊ ?";:INPUT NA$(I):NA$
(I)=LEFT$(NA$(I),8)
200 MN(I)=500
210 NEXT
500 FOR TU=1 TO 6
510 GOSUB1000:GOSUB2000:GOSUB3000
520 NEXT
530 END
1000 FOR I=0 TO 5
1010 OD(I)=RND(9)+1
1020 NEXT
1030 FOR I=0 TO PL
1040 CLS:PRINT"ﾀﾞｲ";TU;"ﾚｰｽ":PRINT"NO.   ﾊﾞｲﾘﾂ"
1050 FOR J=0 TO 5:PRINTJ;":   X";OD(J):NEXT
1060 LOCATE 0,10:PRINTNA$(I);" ｻﾝﾉ ﾊﾞﾝﾃﾞｽ"
1070 PRINT "ｹﾞﾝｻﾞｲ ｼｮｼﾞｷﾝ";MN(I)
1080 INPUT "ﾄﾞﾉｶﾒﾆ ｶｹﾏｽｶ";U(0,I)
1090 IF U(0,I)<0 OR U(0,I)>5 GOTO 1060
1100 INPUT"ｲｸﾗ ｶｹﾏｽｶ";U(1,I)
```

うカメさんもいますが，いつかは全部ゴールインしますから。

全部のカメさんがゴールインしたら，払い戻しです。時々同着の1位がありますが，倍率どおりの払い戻しが当った人みんなにあります。親切なレースですね。

★遊び方

まず最初に遊ぶ人数を入力します。

次に各プレイヤーの名前を入力してください。

名前を入れ終わったら，さっそく第1レースが始まります。1人ずつ順番に好きなカメを選んで番号を入力して，賭け金を入力してください。この時，画面の上に，各カメさんの倍率が表示されます。選んだカメさんが見事1着になれば，この倍率×賭け金が払い戻されます。倍率は最低1，最高9倍までです。賭け金の制限はありません。

全員が賭け終わったら，一斉にスタートです。

カメさんがゴールすると，横に着番が表示され，全部ゴールすると，画面の下の方に

HIT　ANY　KEY

と表示がでますから，好きなキーを押してください。

その後，当たった人に対する配当が表示され，再び

HIT　ANY　KEY

が表示されますから，キーを押してください。次のレースが始まります。レースは全部で6レースあります。

★プログラムの説明

●初期設定　(100〜210)

配列宣言，スプライトの定義，プレイヤーの人数入力，各自名前入力を行ないます。

●メイン　(500〜530)

各レースの処理を呼び出し，6レース行なったら終わりです。

●レース前，カメ選び　(1000〜1140)

各カメの倍率を決め（1000〜1020行），各プレイヤーの入力を行ないます。

まず倍率を表示して(1040〜1050行)，そのプレイヤーの所持金を表示し，選んだカメさんの番号を入力，続けて賭金を入力します（1060〜1130行)。

●レース処理　(2000〜2180)

画面をクリアした後，カメの番号と倍率を表示します。スタート位置にカメをセットして，全部のカメを動かします（2010〜2050)。

後は，各カメを乱数で進めて，ゴールに達すれば順位をGL(H)にセットします。変数GH

```
1110 IF U(1,I)<0 OR U(1,I)>MN(I) GOTO 1100
1120 MN(I)=MN(I)-U(1,I)
1130 NEXT
1140 RETURN
2000 VIEW:GH=0:W$="":SPRITEON
2010 FOR H=0 TO 5
2020 LOCATE 0,H*3+2:PRINT"NO.";H
2030 LOCATE 24,H*3+2:PRINT"X";OD(H)
2040 POSITION H,224,H*24+32
2050 GL(H)=0:NEXT:MOVE0,1,2,3,4,5
2060 R=GH+1
2070 FOR H=0 TO 5
2080 IF GL(H)>0 GOTO 2120
2090 IF MOVE(H) GOTO 2110
2100 IF RND(9)=0 THEN MOVEH
2110 IF XPOS(H)<40 GOSUB 2150
2120 NEXT
2130 IF GH=6 THEN RETURN
2140 GOTO 2060
2150 IF R=1 THEN W$=W$+HEX$(H)
2160 GL(H)=R:GH=GH+1
2170 LOCATE 4,H*3+2:PRINTR
2180 RETURN
3000 LOCATE0,22:PRINT"HIT ANY KEY";:PAUSE:SPRITEOF
F
3010 CLS:FOR I=0 TO PL
3020 FOR J=1 TO 6
3030 IF MID$(W$,J,1)=HEX$(U(0,I)) THEN MN(I)=MN(I)
+U(1,I)*OD(U(0,I)):PRINTNA$(I);" ｻﾝ ﾉ ﾊｲﾄｳ";U(1,I)
*OD(U(0,I))
```

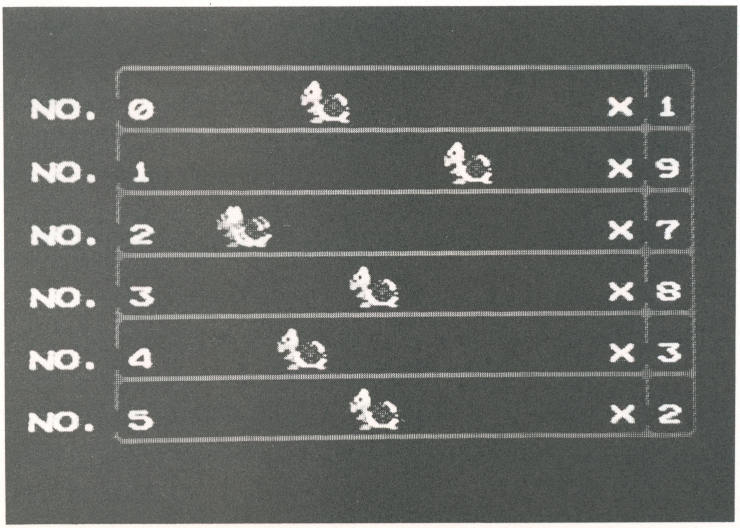
はゴールしたカメの数で，GH＝6になれば処理を終わります。

●レース後処理　(3000〜3060)

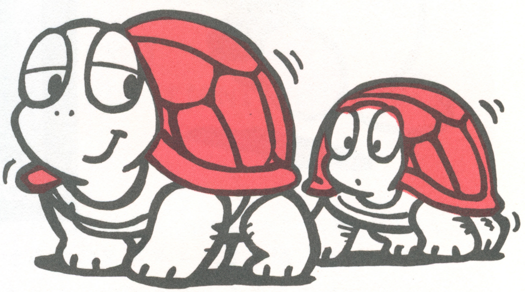
画面をクリアしてから，1番のカメと，各プレイヤーの予想を比較し，当たっていれば所持金に倍率×賭け金を加えます。

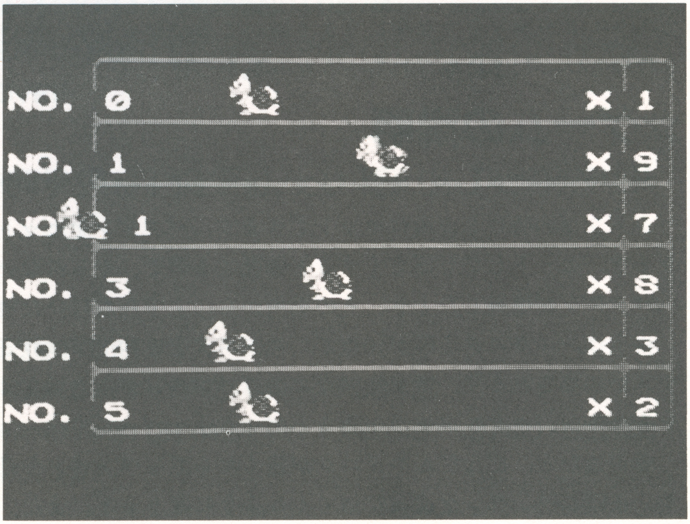
★変数表

OD（H）	カメの倍率
MN（P）	プレイヤー所持金
U(Ø，P)	プレイヤー予想（カメの番号）
U(1，P)	プレイヤー賭け金
NA$(P)	プレイヤー名前
GL(H)	カメ順位
W$	1位のカメ番号
PL	プレイヤー人数
P	プレイヤー識別用変数

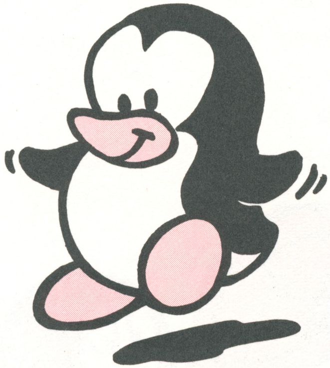


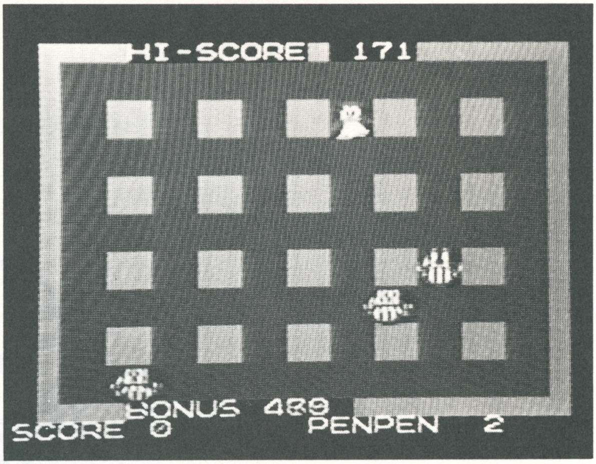
```
3040 NEXT:NEXT
3050 LOCATE0,22:PRINT"HIT ANY KEY";:PAUSE:SPRITEOF
F
3060 RETURN
```





チックタックBOMB





ペンペンがまちがって，ニタニタのすみかに迷いこんでしまいました。３匹のニタニタがペンペンをつかまえようとして近づいてきます。しかたがないので，バクダンでニタニタをやっつけてしまいましょう。でも，ペンペンが持っているのは時限バクダンなんです。だからニタニタが通りそうな所にうまく置いて歩かなくちゃ。置いたらすぐにそこから離れないと自分もいっしょにやられちゃうから気をつけて。

★ゲームの説明

ペンペンを動かして，時限バクダンをうまくセットしてニタニタをやっつけてください。バクダンは同時に３カ所にセットできます。バクダンをセットした所は通路の色が黄色くなります。セットしてしばらくすると爆発するので，やられないようにうまく逃げてください。

バクダンが爆発したとき，その場所から，上下左右にいた物は全部やられてしまいます。

得点は，ニタニタ１匹が10点で，ニタニタを３匹やっつけると１面クリアでその時のボーナスが加算されます。得点2500点ごとにペンペンが１人増えます。全部のペンペンが爆発でやられるか，ニタニタにつかまるとゲームオーバーです。

面が進むと，だんだんニタニタが遠くからでもペンペンを見つけて近づいてくるようになり，正確にペンペンを追っかけてくるようになります。また，３面ごとにニタニタが動くス

プログラムリスト　(BG画面データーは100ページ)

```
100 CGSET1,2:SPRITEON:DIMB(2,2):FORI=1TO4:DEFMOVE(
I)=SPRITE(4,I*2-1,3,8,0,0):NEXT
110 PALET B 1,47,38,38,38:PALET B 2,47,22,22,22
300 LOCATE5,8:PRINT"HIT ANY KEY !";:PAUSE
310 IFSC>HS THENHS=SC
320 SC=0:BN=2500:SP=3:R=100:PE=2
400 IFMOVE(D)GOTO400
410 VIEW:GOSUB3000:GOSUB3100:GOSUB3200:GOSUB3300:E
RAD,5,6,7:E=3:TM=500:D=2:CX=4:CY=1
420 POSITIOND,32,32:FORI=5TO7:POSITIONI,160,128:B(
0,I-5)=0:NEXT:MOVED
500 IFSTRIG(0)=8GOSUB1000
510 S=STICK(0):S=-(S=1)*2-(S=2)*4-(S=4)*3-(S=8)*1
520 IFS=0ORMOVE(D)GOTO570
530 X=CX-(S=2)*2+(S=4)*2:Y=CY-(S=3)*2+(S=1)*2
540 IFASC(SCR$(X+1,Y+1))<>255GOTO570
550 IFD<>S THENERAD:POSITIONS,XPOS(D),YPOS(D):D=S
560 CX=X:CY=Y:MOVED
570 FORI=0TO2
580 IFB(0,I)>0THENB(0,I)=B(0,I)-1:IFB(0,I)=0GOSUB1
500
590 NEXT
600 FORI=5TO7
610 IFXPOS(I)=255GOTO710
620 X=XPOS(I)-XPOS(D):Y=YPOS(I)-YPOS(D)
630 IF(ABS(X)<14ANDY=0)OR(ABS(Y)<14ANDX=0)GOTO2000
640 IFMOVE(I)GOTO710
```

ピードが少しずつ早くなり，９面で最も難しくなって，10面でまた１面に戻ります。

★遊び方

プログラムをRUNしますと

HIT ANY KEY！

と表示されますから，キーボードの好きなキーを（STOPキーはだめですよ!!）押すとゲームが始まります。

ペンペンを動かすにはコントローラーⅠを使います。十字ボタンでペンペンは上下左右に動き，Aボタンで時限バクダンをペンペンの位置にセットします。この時，Aボタンを押しっぱなしにしていると同じ場所に２つのバクダンがセットされてしまう事があるので注意してください。

★プログラムの説明

●初期設定　(100〜420)

100，110行が配列宣言とスプライト，パレットの定義部分です。

300行以降が変数の初期化やスプライト座標セットなどの部分です。ペンペンの数を増やしたい人は，320行の，PE＝２を変更してください。

● ペンペン移動　(500〜560)

まずコントローラーの入力を調べ，Aボタンが押されていれば1000行からのバクダンセット処理を呼びます。

次に方向の入力を調べ，方向（１〜４）を変数Dにセットします。入力があればその方向に移動を行ないます。入力がない時は，ペンペンが移動中の時はここの処理はスキップします。

●時限バクダンのカウント　(570〜590)

３つのバクダンを調べ，セットされていればタイマーを１減らします。０になったら1500行からの爆発処理を呼びます。

●ニタニタ移動

ペンペンとの距離を調べ，それによって移動方向を決め，移動します。

ペンペンをつかまえたかどうかは630行で両方の座標を調べ，重なっていたら2000行に飛びます。

710行では，ボーナスを１点減らします。

●時限バクダンセット　(1000〜1020)

バクダンが３つともセットされていたら（全部のタイマーが０でなければ），何もしないで

```
650 L=ABS(X)+ABS(Y):IFRND(R)<L THENH=RND(4)*2+1:GO
TO690
670 IFABS(X)>ABS(Y)THENH=3-(X>0)*4:GOTO690
680 H=1-(Y<0)*4
690 X1=XPOS(I):X=(X1-16)/8-(H=3)*2+(H=7):Y1=YPOS(I
):Y=(Y1-24)/8-(H=5)*2+(H=1)
700 IFASC(SCR$(X,Y))=255THENDEFMOVE(I)=SPRITE(11,H
,SP,16,0,2):POSITIONI,X1,Y1:MOVEI
710 NEXT:IFTM>0THENTM=TM-1:GOSUB3300
730 GOTO500
1000 IFB(0,N)>0THENRETURN
1010 B(1,N)=CX:B(2,N)=CY:B(0,N)=15
1020 COLORCX,CY,1:N=(N+1)MOD3:RETURN
1500 X=B(1,I):Y=B(2,I):FORJ=2TO0STEP-2
1510 COLORX,Y,J:COLORX+2,Y,J:COLORX-2,Y,J:COLORX,Y
-1,J:COLORX,Y+2,J:NEXT:X=X*8+16:Y=Y*8+24
1520 FORJ=5TO7:X1=XPOS(J):Y1=YPOS(J)
1530 IF NOT((ABS(X1-X)<24 AND Y1=Y)OR(ABS(Y1-Y)<24
 AND X1=X))GOTO1560
1540 ERAJ:POSITIONJ,255,255:E=E-1:IFE=0RETURN2500
1550 SC=SC+10:GOSUB3000:IFSC>=BN THENPE=PE+1:GOSUB
3200:BN=BN+2500
1560 NEXT:X=B(1,I):Y=B(2,I)
1570 IF(ABS(CX-X)<3ANDCY=Y)OR(ABS(CY-Y)<3ANDCX=X)R
ETURN2000
1590 RETURN
2000 PLAY"C1ECG":PE=PE-1:IFPE<0GOTO300
2010 GOTO400
2500 SC=SC+TM:GOSUB3000:R=R+25:IFR<175GOTO400
2520 R=100:SP=SP-1:IFSP=0THENSP=3
```

戻ります。そうでなければ，バクダンのX座標を配列B（1，N）に，Y座標をB（2，N）にセットし，爆発までのタイマーとしてB(0，N)に時間（今は15）をセットします。

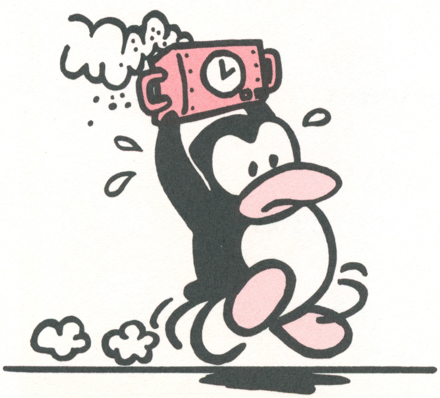
時間を短かくしたい人は，この数字を小さくしてください。

●バクダン爆発処理　（1560〜1590）

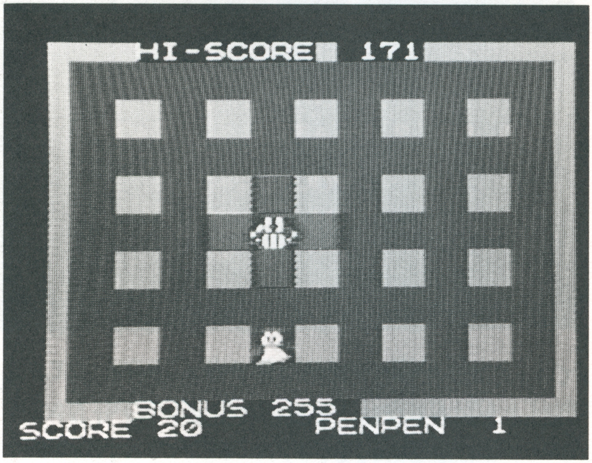
まずその座標のまわりにニタニタがあるかを調べ，いたら消して得点を10点加算します。ニタニタの数が変数Eにセットされているので，これを1減らします。Eが0になったら1面クリアですから2500行に飛びます。次にペンペンの座標を調べ，爆発の範囲に入っていたら2000行に飛びます。

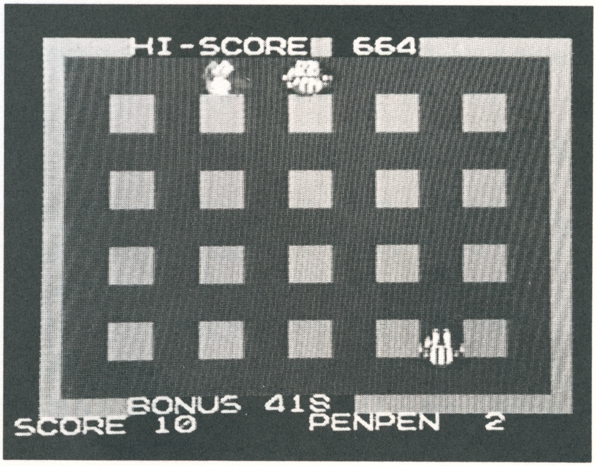
★変数表

		スプライト
CX，CY	ペンペンの座標	1〜4 ペンペン
D	ペンペンの向き	5〜7 ニタニタ
N	バクダンの番号	
B(0，N)	バクダンのタイマー	
B(1，N)	バクダンのX座標	
B(2，N)	バクダンのY座標	
E	ニタニタの数	
PE	ペンペンの数	
TM	ボーナス	
SP	ニタニタのスピード	

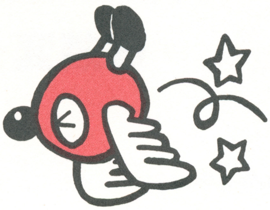


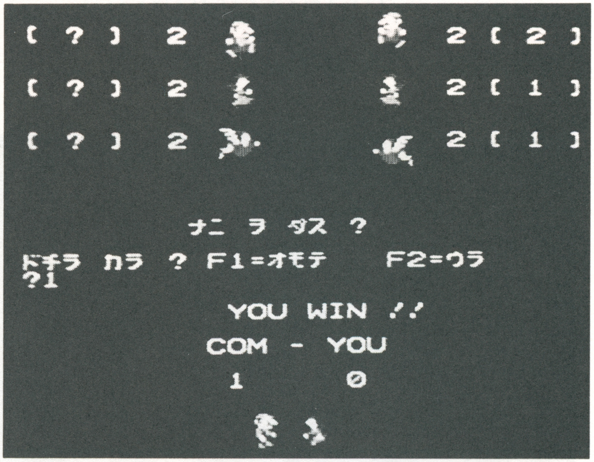
```
2530 GOTO400
3000 LOCATE5,20:PRINTSC:RETURN
3100 LOCATE14,0:PRINTHS:RETURN
3200 LOCATE20,20:PRINTPE;"  ":RETURN
3300 LOCATE10,19:PRINTTM;"  ":RETURN
```





マリオでジャンケン





ジャンケンは日本のふるーいあそびです。でも，日本だけのあそびではありません。「グー」「チョキ」「パー」こそ使いませんが，同じようなあそびは，世界中にあるそうです。

グー　（石）　　紙には包まれるがハサミには切られない

チョキ(ハサミ)　紙は切れるが石は切れない

パー　（紙）　　ハサミには切られるが石を包む

ジャンケンの意味は上の通りですが，他のものを使ったもの（猟師と鉄砲と熊，へびとかえるとなめくじ）もたくさんあるようです。

★ゲームの説明

このファミコン版ジャンケンは次の三つを使います。

マリオ　——アキレスには強いがレディに弱い

レディ　——マリオには女の強味，でもアキレスには負ける

アキレス——力ではマリオに負けるけどレディには勝つ

だけど，ただのジャンケンではつまらない。そこで，次のようなルールを決めました。

コンピューターもプレイヤーも，マリオ・レディ・アキレスのカードをそれぞれ２枚持つ。この６枚は表にして相手に見せます（オモテカード）。そのあと５枚のカードが配られます。

プログラムリスト

```
10 CLS:SPRITE ON:DIM A(1,5)
20 FOR I=0 TO 2:READ A,J
30 DEF SPRITE I,(I,1,0,0,0)=CHR$(A)+CHR$(A+1)+CHR$
(A+2)+CHR$(A+3)
35 DEF MOVE(I)=SPRITE(J,7,1,20,0,I)
40 DEF SPRITE 3+I,(I,1,0,1,0)=CHR$(A+1)+CHR$(A)+CH
R$(A+3)+CHR$(A+2)
45 DEF MOVE(I+3)=SPRITE(J,3,1,20,0,I)
50 KEY I+1,CHR$(49+I)+CHR$(13):NEXT
60 DATA 0,0,28,1,64,3
100 FOR I=0 TO 2
110 SPRITE I,154,24+I*28:SPRITE I+3,92,28+I*24
120 A(0,I)=2:A(1,I)=2
130 NEXT:CV=0:YV=0:CY=0
140 FOR I=0 TO 4:J=RND(3)+3:A(0,J)=A(0,J)+1:J=RND(
3)+3:A(1,J)=A(1,J)+1:NEXT
200 GOSUB 900
210 LOCATE 8,12:PRINT "ナニ ヲ ダス ?";
220 IF A(1,0)+A(1,3)>0 THEN LOCATE 0,14:PRINT "F1=
マリオ  ";
230 IF A(1,1)+A(1,4)>0 THEN LOCATE 9,14:PRINT "F2=
レデイ";
240 IF A(1,2)+A(1,5)>0 THEN LOCATE 18,14:PRINT "F3
=アキレス"
250 LOCATE 0,15:PRINT "      ";:LOCATE 0,15:INPUT
K:IF K<1 OR K>3 THEN 250
260 IF A(1,K-1)+A(1,K+2)<1 THEN 250
270 IF A(1,K-1)=0 THEN 350
```

これは何がくるかまったくわかりません。そして この5枚のカードは相手にみせません（ウラ・カード)。この合計11枚のカードを1枚ずつ出し合って勝負を決めます。

★遊び方

プログラムを打ちこんだあと「RUN」すると，**図-1** のようになります。①の数字がオモテカードの枚数，②はウラカードの枚数です。

「F 1」「F 2」「F 3」を使って出すカードの種類を決めます。そのカードがない時は受けつけません。そのカードが，オモテとウラと両方持っている時には，どちらから出すかを聞いてきますので，「F 1」「F 2」を使って決めてください。

カードが決定すると，両方から選んだカードが走り出してきます。

コンピューター側のオモテカードを見ながら作戦をたててください。たとえば，のこり4枚でオモテカードが4枚しかなければ当然ウラカードにのこりはありません。やり方しだいでは全勝も可能です。

★プログラムの説明

30～ 60	マリオ・レディ・アキレスのスプライトとムーブを指定しています。
	DEF SPRITE ／ DEF MOVE ｝ 0～2左むき ／ 3～5右むき
100～140	スプライトを表示
	各カード初期設定
200～	メインルーチン
200～280	カードの種類を決定
300～400	ウラかオモテかを決定
	それぞれのカードを減らす
450～500	コンピューターのカード決定
550～700	選んだキャラクター移動
	勝敗を決定
	回数が11回以下なら200行へ
800～899	総合判定
900～910	残り枚数の表示と現在までの勝負の表示のサブルーチン

ところがこのプログラム，現在ではコンピューター側のカード選択は乱数で行なわれています。2Kのメモリーではぎりぎりの限　です。

Ver. 3を持っている方，あるいは，マリオやレディのキャラクターを使わなくても（見た

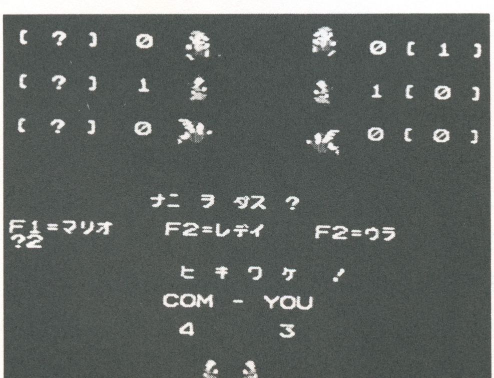
```
280 IF A(1,K+2)=0 THEN 400
300 LOCATE 0,14:PRINT "ト゛チラ カラ ? F1=オモテ   F2=ウラ
 ";
310 LOCATE 0,15:PRINT "          ";:LOCATE 0,15:INPUT
H:IF H<1 OR H>2 THEN 310
320 IF H=1 THEN 400
350 A(1,K+2)=A(1,K+2)-1:GOTO 450
400 A(1,K-1)=A(1,K-1)-1
450 CK=RND(6):IF A(0,CK)<1 THEN 450
500 A(0,CK)=A(0,CK)-1:IF CK>2 THEN CK=CK-3
550 K=K-1:MOVE (K):MOVE(CK+3)
600 PAUSE 100:LOCATE 10,17:IF K=CK THEN PRINT "ヒ キ
 ワ ケ":CY=CY+1:GOTO 700
610 IF (K=0 AND CK=1) OR (K=1 AND CK=2) OR (K=2 AN
D CK=0) THEN PRINT "COMPUTER !":CV=CV+1:GOTO 700
620 IF (K=0 AND CK=2) OR (K=1 AND CK=0) OR (K=2 AN
D CK=1) THEN PRINT "YOU WIN !":YV=YV+1
700 PAUSE 150:ERA K,CK+3:LOCATE 8,17:PRINT "
    ";:IF CV+YV+CY<11 THEN 200
800 SPRITE OFF:LOCATE 0,23:PRINT:PRINT:PRINT:PRINT
:PRINT:PRINT:PRINT
810 IF CV>YV THEN PRINT "COMPUTER WIN ";:GOTO 899
820 IF YV>CV THEN PRINT " Y O U   WIN ";:GOTO 899
830 PRINT " ヒ キ ワ ケ";
899 PRINT:PRINT:END
900 FOR I=0 TO 2:LOCATE 0,1+I*3:PRINT "[ ? ] ";A(0
,I);:LOCATE 20,1+I*3:PRINT A(1,I);" [";A(1,I+3);"
]";:POSITION I,164,210:POSITION I+3,62,210:NEXT
910 LOCATE 9,19:PRINT "COM - YOU";:LOCATE 9,21:PRI
NT CV;"    ";YV;:RETURN
```

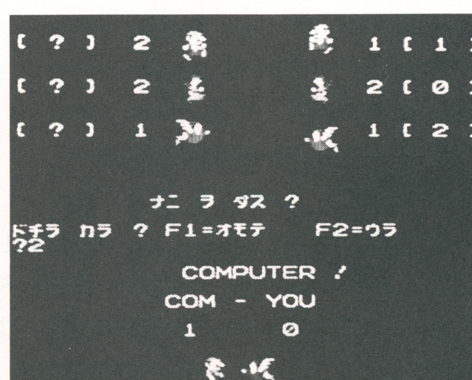
目が悪くても)，コンピューターをもう少しお利口さんにしたい場合は，450行を変えてください(**リスト-2**)。

また，最初に配られるウラカードを自分で選べるようにするのもおもしろいでしょう(**リスト-3**)。

★変数表

A(Ø，Ø)	コンピューター表マリオ
A(Ø，1)	コンピューター表レディ
A(Ø，2)	コンピューター表アキレス
A(Ø，3)	コンピューター裏マリオ
A(Ø，4)	コンピューター裏レディ
A(Ø，5)	コンピューター裏アキレス
└→1だとコンピューター	
K	あなたの出すカード
CK	コンピューターの出すカード
CV	コンピューターの勝ち
YV	あなたの勝ち
CY	引き分け





プログラムリスト2

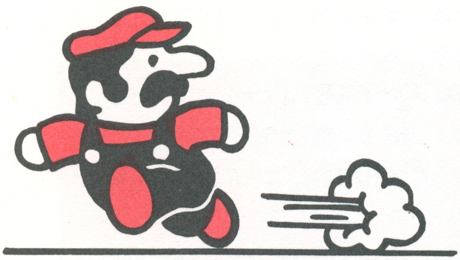
```
460 DK=CK:IF DK>2 THEN DK=DK-3
470 IF RND(3)=0 OR CV+YV+CY>9 THEN 500
480 DU=DK+1:IF DU=3 THEN DU=0
490 IF A(1,DU)+A(1,DU+3)>=(11-CY+CV+YV)/3 THEN 450
```

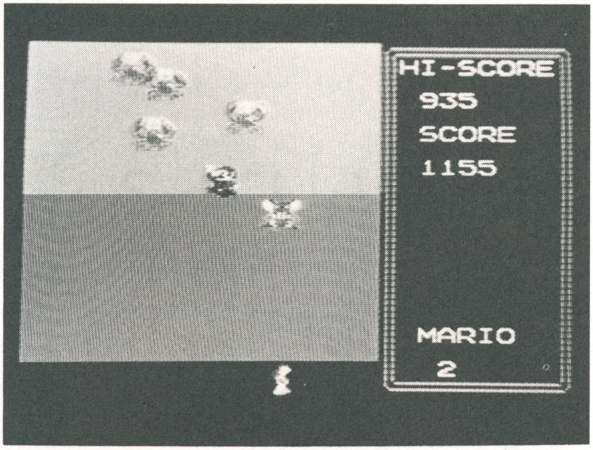
プログラムリスト3

```
140 FOR I=0 TO 4:J=RND(3)+3:A(0,J)=A(0,J)+1
150 LOCATE 3,12:PRINT I+1;"ﾏｲﾒ.  ﾄﾞﾚ ｦ ｴﾗﾌﾞ ?";
160 LOCATE 0,14:PRINT "F1=ﾏﾘｵ   F2=ﾚﾃﾞｨ   F3=ｱｷﾚｽ"
;
170 LOCATE 0,15:PRINT "       ";:LOCATE 0,15:INPUT
J:IF J<1 OR J>3 THEN 170
180 A(1,J+2)=A(1,J+2)+1
190 NEXT:LOCATE 0,12:PRINT "
   ";:LOCATE 0,14:PRINT "
";:LOCATE 0,15:PRINT "       ";
```



MARIO KICK





「ねえ，マリオ。今日の晩ごはん何にしましょうか？」
「うーん，そうだな。カニが食べたいな」
「あ，それいいわね。でもカニは高いのよねー。そうだ，マリオ，ちょっと浜まで行ってカニを採って来てよ」
「メンドクサイナア。じゃ別に今日はカニじゃなくてもいいや」
「ダメダメ。マリオが言いだしたんだから。たくさん採ってこないと晩ごはん抜きよ」
「そんなー」

ある日の夕方，マリオとレディーの間で上のような会話がありまして，結局マリオはカニを採りに行かされそうです。でも本当はあんまり行きたくないみたい。あっちへ行ったり，こっちへ行ったり，ふらふら歩いています。どうやら，なんとかして逃げだそうとしているみたい。

でも，ここでマリオに逃げられたらカニが食べられません。そこで，カニを採ってトコトコ帰ってくるマリオに，さらに採りに行ってもらいましょう。

あっ，ブンブン飛んでくるハエさんには気をつけて。レディーはハエが大嫌い。さわっただけで足がすくんでしばらく動けなくなってしまいます。ほらほら，その間にマリオが逃げちゃうよ。

プログラムリスト　（BG画面データーは101ページ）

```
100 CLS:CGEN2:CGSET1,0:SPRITEON:VIEW
110 DEFMOVE(7)=SPRITE(2,5,1,96,0,1)
120 DEFMOVE(1)=SPRITE(1,3,1,8,0,2)
130 DEFMOVE(2)=SPRITE(1,7,1,8,0,2)
140 PALETS 1,47,48,21,18
300 LOCATE2,8:PRINT"PUSH START KEY!"
310 IFSTRIG(0)<>1GOTO310
320 ML=2:SC=0:B=1000:VIEW:GOSUB3000:GOSUB3100:GOSU
B3200
400 X=RND(129)+16:Y=RND(50)+100:M=RND(2)*6+2:ERA1,
2
410 W=0:C=0:D=1:GOSUB1110
420 POSITION1,16,176:MOVE1
430 FORI=3TO6:POSITIONI,0,0:ERAI:NEXT
500 S=STICK(0):IFS=0ORS>2ORW>0ORMOVE(D)GOTO530
510 IFD<>S THENERAD:POSITIONS,XPOS(D),176:D=S
520 IFNOT((D=1ANDXPOS(D)=144)OR(D=2ANDXPOS(D)=16))
THENMOVED
530 IFMOVE(0)GOTO570
540 IFYPOS(0)>174THENPOSITION0,XPOS(0),176:GOTO250
0
550 IFYPOS(0)<26THENPOSITION0,XPOS(0),24:J=0:GOSUB
1100:GOTO570
560 GOSUB1000
570 IFABS(XPOS(D)-XPOS(0))<16ANDABS(YPOS(D)-YPOS(0
))<16 GOSUB2000
580 FORI=3TO6
```

★遊び方

リストを全部入力できたらRUNしましょう。

PUSH START KEY!

と表示がでましたか？　でたらコントローラーIのスタートボタンを押してください。そうするとゲームが始まります。

画面の下にレディーが表示されましたね。このレディーはコントローラーIの十字ボタンを使って左右に動かせます。

あとはブロックくずしの要領です。上の方で動いているカニをマリオが採ってきたら，レディーをうまく動かしてハネ返してください。ブロックくずしと違って，カニを採ってきたマリオをハネ返さないと得点は増えません（そうしないとカニが受取れないでしょ？）。点数はカニ1匹につき5点です。

レディーがハエにふれるとしばらく動けなくなります。

マリオに3回逃げられるとゲームオーバーです。

★プログラムの説明

プログラムの構成は次のようになっています。

100〜	初期設定
500〜	メインルーチン
1000〜	マリオ反射
1500〜	カニ採集
2000〜	カニ受取り
2500〜	マリオ逃げる
3000〜	得点などの表示用
4000〜	GAME OVER

以上のような処理の中で一番重要でややこしいマリオ反射の部分を中心に説明します。

壁やカニにマリオがぶつかると，光と同じように反射します(図参照)。左右の壁で反射する場合の処理が1000〜で，上の壁やカニで反射する場合は1100〜の処理を行ないます。マリオの移動方向は変数Mに入っていますから，反射したあとの方向は1000行や1100行のような論理式を使った式で計算できます。問題は，反射してから次の壁にぶつかるまでの移動距離で，これは簡単な計算式にはできないので，1010〜1050行や，1110〜1155行のように方向別に計算して，変数Lにセットします。MとLが決まったら1160行で新しい移動方向と距離を定義して，1170行で動かします。

```
590 IFABS(XPOS(I)-XPOS(0))<16ANDABS(YPOS(I)-YPOS(0
))<16 THENPLAY"C0":GOSUB1500
600 IFXPOS(I)=0GOSUB1550:GOTO620
610 IF NOT MOVE(I) THENERAI:POSITIONI,0,0
620 NEXT
630 IFABS(XPOS(D)-XPOS(7))<16ANDABS(YPOS(D)-YPOS(7
))<16THENW=W+25:POSITION7,255,255
640 IFNOT MOVE(7)THENPOSITION7,RND(129)+16,0:MOVE7

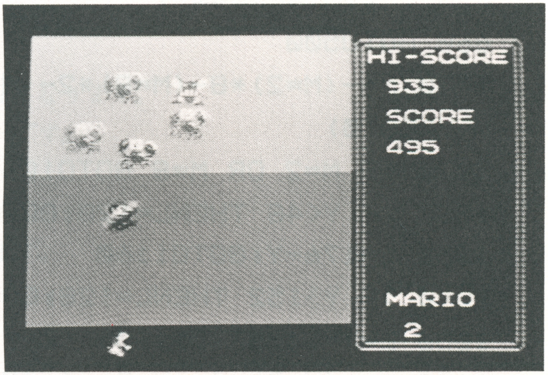
650 IFW>0THENW=W-1
660 GOTO500
1000 M=-(M=2)*8-(M=8)*2-(M=4)*6-(M=6)*4:X=XPOS(0):
Y=YPOS(0)
1010 IF M=2 OR M=8 GOTO1040
1020 IF Y<48 THENL=128:GOTO1160
1030 L=176-Y:GOTO1160
1040 IF Y>152 THENL=128:GOTO1160
1050 L=Y-24:GOTO1160
1100 M=-(M=2)*4-(M=8)*6-(M=6)*8-(M=4)*2:X=XPOS(0):
Y=YPOS(0)
1110 ON M/2 GOTO1120,1130,1140,1150
1120 IF 144-X<Y-24 THEN L=144-X:GOTO1156
1125 L=Y-24:GOTO1156
1130 IF 144-X<176-Y THEN L=144-X:GOTO1156
1135 L=176-Y:GOTO1156
1140 IF X-16<176-Y THEN L=X-16:GOTO1156
1145 L=176-Y:GOTO1156
1150 IF X-16<Y-24 THEN L=X-16:GOTO1156
1155 L=Y-24
1156 IFL<2THENM=-(M<5)*6-(M>5)*4:GOTO1110
```

メインルーチンは，キー入力とレディーの移動（500～520），マリオの状態チェック（530～570），カニ移動（580～620），ハエ移動（630～640）のようになっています。

それぞれの移動処理の中で他のスプライトとの衝突を調べ，意味のある衝突であれば各処理を行ないます。この処理は，1500（マリオとカニ），2000（マリオとレディー），630（ハエとレディー））の3種類です。

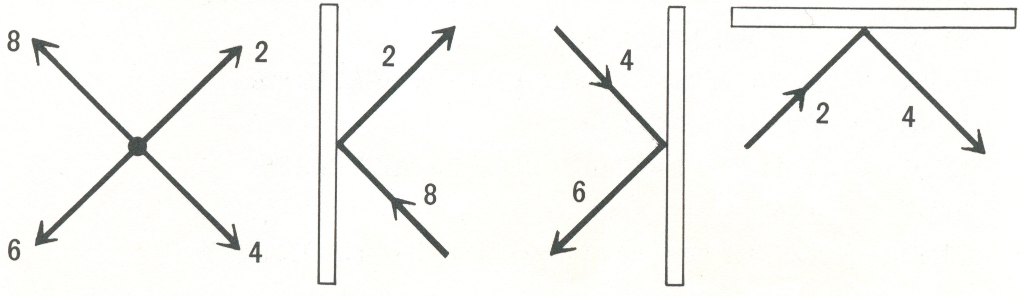
★変数表

D	レディの移動方向
M	マリオの移動方向
L	マリオの移動距離（ドット）
C	マリオが持っているカニの数
W	レディが動けない時間
ML	マリオの数
SC	スコア
HS	ハイスコア

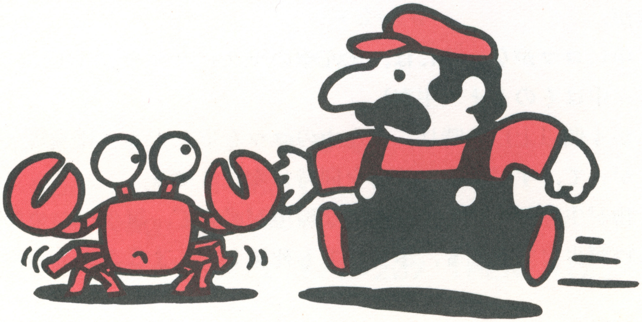


スプライト		定義行
0	マリオ	1170行
1，2	レディ	120～130行
3～6	カニ	1550行
7	ハエ	110行

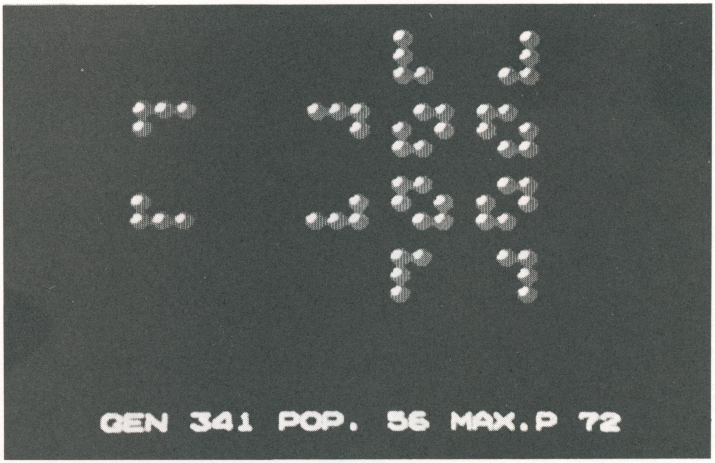
マリオの移動方向



```
1160 DEFMOVE(0)=SPRITE(0,M,1,L/2,0,0)
1170 POSITION0,X,Y:MOVE0:RETURN
1500 IF I=J THEN RETURN
1510 C=C+1:GOSUB1100
1520 J=I:ERAI:POSITIONI,0,0:RETURN
1550 H=RND(2)*4+3:IFRND(30)>0THENRETURN
1560 DEFMOVE(I)=SPRITE(14,H,RND(3)+1,64,0,3)
1570 POSITIONI,-(H=3)*16-(H=7)*144,RND(57)+24:MOVE
I:RETURN
2000 IF I=J THEN RETURN
2010 J=I:GOSUB1100:IFC=0THENRETURN
2020 SC=SC+C*5:GOSUB3000:C=0
2030 IFSC>=B THENML=ML+1:B=B+1000:GOSUB3100
2040 RETURN
2500 PLAY"C0DCD":ML=ML-1:GOSUB3100
2510 IFML<0GOTO4000
2520 GOTO400
3000 LOCATE19,7:PRINTSC:RETURN
3100 LOCATE20,19:PRINTML;" ":RETURN
3200 LOCATE19,3:PRINTHS:RETURN
4000 LOCATE5,5:PRINT"GAME OVER"
4010 IFSC>HS THENHS=SC:GOSUB3200
4020 GOTO300
```



ライフゲーム



ライフゲームというのは15年ほど前にアメリカのコンウェー博士が考案したもので，ゲームというよりはシミュレーションに近いものです。これは，あるルールを定め，そのルールのもとで生物が誕生，生存，死滅するのをシミュレートするものです。特にコンピューター用として作られたものではないので，コンピューターなしでも楽しむ事ができます。

ゲームはマス目の上で行ないます。マス目の大きさは理論的には無限大でなくてはいけませんが，実現不可能ですから好きな大きさを決めます。後で説明するパターンをすべて楽しむには最低でも100×100のマスが必要だと言われています。

ライフゲームのルールは次の４つだけです。

まず，ある１つのマス目に関して，その周囲の８個のマスにコマ（生物を表わします）があるかを調べ，コマの数が，

①３個で，かつ，まん中にコマがない時には，まん中のマスにコマを置きます（誕生)。

②２，３個ならば，まん中はそのまま（生存)。

③それ以外（１個以下か４個以上）の時は，まん中のコマを取り去る（死滅)。

④以上のルールにしたがって，すべてのマス目を同時に変化させる。

①～③のルールは，生物が生存していくには適度な環境が必要で，誕生のための条件はさらにきびしく，また，生物が少なすぎても，多すぎても死滅してしまうことを表わしている。

プログラムリスト

```
100 CGEN 3:SPRITE ON:CLS
110 DEF SPRITE 0,(0,0,0,0,0)="+"
120 PALET B 0,47,48,33,22
130 PRINT "ﾊﾞﾝ ﾉ ｵｵｷｻ ｦ ｷﾒﾃ ｸﾀﾞｻｲ"
140 INPUT "ﾖ ｺ (3-28)";MX
150 IF MX<3 OR MX>28 GOTO 140
160 INPUT "ﾀ ﾃ (3-24)";MY
170 IF MY<3 OR MY>24 GOTO 160
180 DIM A(MX-1,MY-1)
500 X=MX/2:Y=MY/2:CLS
510 SPRITE 0,X*8+16,Y*8+24
520 S=STRIG(0)
530 IF S=1 GOTO 1000
540 IF S=4 THEN LOCATE X,Y:PRINT CHR$(207);:GOTO 5
60
550 IF S=8 THEN LOCATE X,Y:PRINT " ";
560 S=STICK(0)
570 IF S=1 THEN X=(X+1) MOD MX
580 IF S=2 THEN X=(X-1+MX) MOD MX
590 IF S=4 THEN Y=(Y+1) MOD MY
600 IF S=8 THEN Y=(Y-1+MY) MOD MY
610 SPRITE 0,X*8+16,Y*8+24
620 FOR I=0 TO 200:NEXT
630 GOTO 520
1000 SPRITE OFF:P=0:G=0
1010 FOR X=0 TO MX-1
1020 FOR Y=0 TO MY-1
1030 IF ASC(SCR$(X,Y))=207 THEN A(X,Y)=1:P=P+1
```

このルールさえ守ればコンピューターなしでも実行できます。ためしにやってみるには碁盤かオセロボードが便利です。コンピューターを使わない時に問題になるのは，マス目の数と条件の④です。実際にやってみるとわかりますが，順番に変化させていけないので，まずすべてのマスをチェックして誕生，死滅をマークしていき，チェックがすべて終わってから変化させなくてはならないわけですから，マスが20×20位になるとかなり大変になり，100×100なんてやる気もおこりません。

実は，これはコンピューターを使ってもわりと大変で，特にベーシックを使うとスピードの問題があるのですが，まあながめていればいいわけですから人力よりはるかに楽といえます。しかし今度は広さの問題がでてきます。ファミリーベーシックを使うとマス目は最大でも28×24しかとれませんから，この範囲で楽しめる事を考えなくてはなりません。

ライフゲームの楽しみは，広い範囲にコマをバラまいて，時間がたつにつれてコマが繁殖していったり，突然減少してしまったりする変化を見る事の他に，色々なパターンを発見する事にあります。これはアメリカではかなり研究されていて，色々とおもしろいパターンが知られていますので，その中からいくつかを紹介しておきましょう。

パターンはいくつかに分類でき，まったく変化しないものや，一定の周期を持って同じ変化をくり返すもの，パターンがだんだん移動していくものなどがあります。中にはふたつのパターンの間を別のパターンが往復するというすごいパターンも研究されています。

★プログラムの使い方

プログラムが始まると，まずマス目の大きさを入力します。ヨコ（3～20マス）タテ（3～20マス）以内で入力します。必要もないのに大きくすると，その分遅くなりますから，必要最小限の大きさにしましょう。たとえば「ブリンカー」を見たいならば5×5マスで十分です。

次は，パターンのセットです。画面に“＋”がでますが，これがカーソルです。コントローラーⅠでカーソルを動かし，Bボタンを押すとコマがセットされます。間違った時はAボタンで消すことができます。

パターンがセットできたらSTARTボタンを押してください。あとはパターンの変化をながめていましょう。

画面の下に出ているのは，

GEN	何周期目か
POP	入口，コマの数です
MAX P	それまでの最大入力です

やめたい時はSTOPキーを押してください。

```
1040 NEXT:NEXT:MP=P
1050 FOR CX=0 TO MX-1
1060 FOR CY=0 TO MY-1
1070 N=0
1080 FOR I=-1 TO 1
1090 X=(CX+I+MX) MOD MX
1100 FOR J=-1 TO 1
1110 Y=(CY+J+MY) MOD MY
1120 IF ASC(SCR$(X,Y))=207 THEN N=N+1
1130 NEXT:NEXT
1140 IF N=3 THEN A(CX,CY)=1:GOTO 1160
1150 IF N<>4 THEN A(CX,CY)=0
1160 NEXT:NEXT
1170 P=0
1180 FOR X=0 TO MX-1
1190 FOR Y=0 TO MY-1
1200 IF A(X,Y)=0 THEN LOCATE X,Y:PRINT " ";:GOTO 1
220
1210 LOCATE X,Y:PRINT CHR$(207);:P=P+1
1220 NEXT:NEXT:G=G+1
1230 IF P>MP THEN MP=P
1240 LOCATE 0,23:PRINT "GEN";G;" POP.";P;" MAX.P";
MP;
1250 GOTO 1050
```

★プログラムの説明

●100〜180行

初期設定です。マス目の大きさもここで入力します。ヨコの大きさは変数MX，タテはMYにセットします。

●500〜630行

コマのセット部です。コントローラーからの入力を調べ，それにしたがってカーソルを動かしたり，コマのセット，消去を行ないます。

●1000〜1040行

ゲームの準備です。配列A（X，Y）はゲーム盤のコピー，座標（X，Y）にコマがあれば1，なければ0にセットされます。変数Pは全体のコマの数（人口）です。

●1050〜1160行

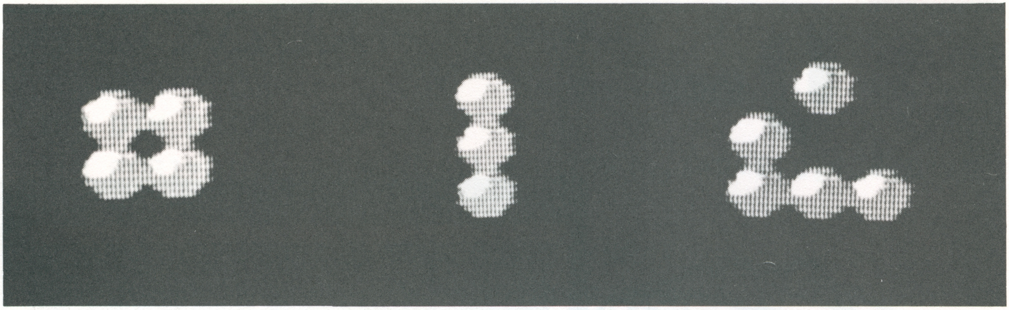
マス目のチェックです。SCR$でBG画面を直接調べ，変化がある場合は配列Aの方を変えます。

あるマス目（CX，CY）の周囲8マスのチェックをしているのは1070〜1130行で，1090，1110行で調べる座標の計算にMODを使うことによって，ゲームの盤の上下と左右をつないでいます。

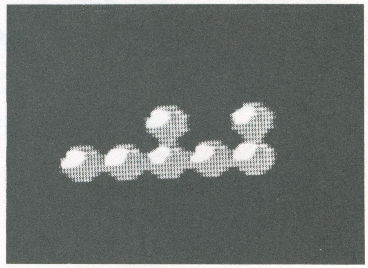
●1170〜1220行

配列Aの内容をBG画面にコピー（表示）します。

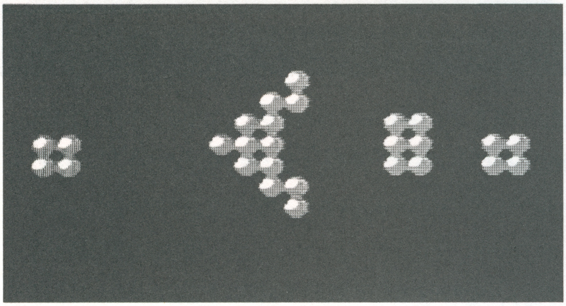
以上（1050行以降）をひたすらくり返すわけです。



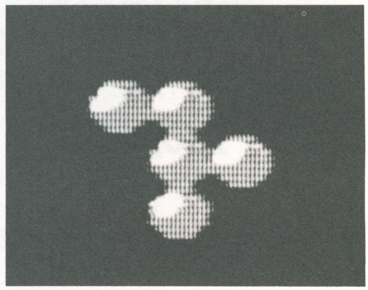
ブロック　　ブリンカー　　グライダー



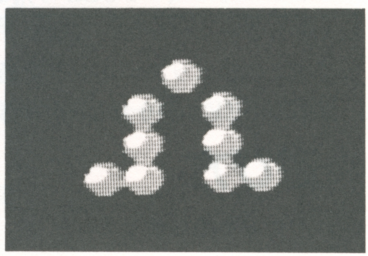
glider generator
グライダー発生器



gueen bee (Shuttle)

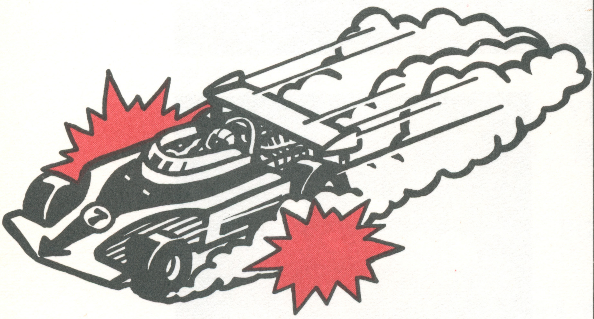


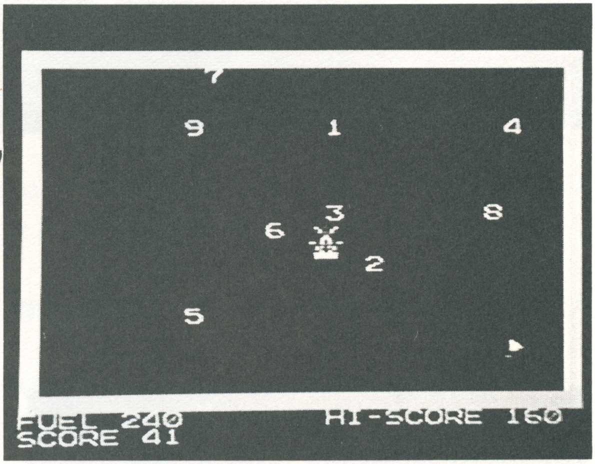
下型ペントミ



hat

ラリー





よし，次は第3チェックポイントだ。ちょっと遠いな。

オット，スペシャルフラッグだ!! しかし方向が完全に反対だなあ。取りに行くべきかどうか…。あーあ，考えていたら消えちゃった。残念!!

★ゲームの説明

コントローラーを使って運転して，0～9までのチェックポイントを順番に通過してください。各チェックポイントを通過すると1ケ所につき10点加算されます。

時々画面にあらわれるハタはスペシャルフラッグで，チェックポイントとは関係なく，通過するとその時の残りFUELの1/10がボーナスとして加算されます。これは一定の時間がたつと消えてしまいますから，すばやく取りに行ってください。

すべてのチェックポイントを通過するとゴールイン。残りのFUELが得点に加算されます。

途中でFUELが0になると燃料切れでゲームオーバーです。

★遊び方

プログラムを走らせると，

PUSH　START　BUTTON!!

プログラムリスト　（BG画面データーは102ページ）

```
100 DIM DX(7),DY(7)
110 FORI=0TO3:DEFMOVE(I)=SPRITE(6,I*2+1,2,4,0,0):N
EXT
120 FORI=0TO7:READDX(I),DY(I):NEXT
130 DATA 0,-1,1,-1,2,0,2,1,1,2,0,2,-1,1,-1,0
140 CGSET1,1:HS=0:SPRITEON:CLS
200 SC=0:FU=500:FL=0:N=0:H=0:OH=0:L=FU-50
210 CX=12:CY=12:POSITION H,112,120
220 LOCATE5,10:PRINT"PUSH START BUTTON !"
230 IF STRIG(0)<>1 GOTO230
250 VIEW:FORI=0TO9
260 X=RND(26)+1:Y=RND(19)+1
270 IFSCR$(X,Y)<>" "GOTO260
280 LOCATEX,Y:PRINTHEX$(I):NEXT
290 LOCATE15,21:PRINT"HI-SCORE";HS
500 S=STRIG(0)/4:H=(OH+(S=1)-(S=2)+4)MOD4:IFMOVE(O
H)GOTO500
510 A=ASC(SCR$(CX+DX(H*2),CY+DY(H*2))):B=ASC(SCR$(
CX+DX(H*2+1),CY+DY(H*2+1)))
520 IF A=32 AND B=32 GOTO580
530 IF A=253 OR B=253 GOTO 620
550 IF (A=32 AND B=199) OR (A=199 AND B=32) GOSUB1
100:GOTO580
560 IF (A-48=N AND B=32) OR (A=32 AND B-48=N) GOSU
B1200:GOTO580
565 IF A-48<>N OR B-48<>N GOTO620
570 GOSUB1200:GOSUB1100
580 IFMOVE(OH)GOTO580
```

と表示が出ますから，コントローラーのスタートボタンを押してください。

画面に０～９の数字が表示されますが，これがチェックポイントです。

同時に車も表示され動きだしますから，この車をコントローラーを使って運転してください。

運転のしかたは，コントローラーのＡ，Ｂのボタンを使い，Ａボタンを押すと車は右にまがり，Ｂボタンを押すと左にまがります。

ゴールインするか，時間切れでゲームオーバーになると，画面にあなたの得点が表示され，再び最初の画面に戻りますから，続けて遊ぶ場合はスタートボタンを押してください。

★プログラムの説明

●初期設定　(100～290)

画面の初期設定や，スプライトの定義，データの読み込み，変数の初期化などを行ないます。燃料の量を変えたい人は200行のＦＵ＝500を変えてください。

チェックポイントのセットは，250～280行です。

●メンルーチン　(500～650)

コントローラーからの入力チェックと車の移動を行ないます。

車の進行方向（０～３）は変数ＯＨにセットされています。これに対して新しい進行方向は変数Ｈにセットします。Ａ，Ｂどちらのボタンも押されていなければＨ＝ＯＨです。Ｈが決まったらその方向の前方の座標に何があるかを調べます。つまり車の進路上に何があるかを調べて，そのコードを変数Ａ，Ｂにセットします(510行)。調べる座標は当然Ｈの値によって違いますから，車の現在位置(ＣＸ，ＣＹ)を中心に，どこを調べればいいかは配列ＤＸ(Ｈ)，ＤＹ(Ｈ)とＤＸ(Ｈ＋１)，ＤＹ(Ｈ＋１)にセットされています。これによってＨに関係なく調べる座標は，

CX＋DX(H)，CY＋DY(H)　と，
CX＋DX(H＋1)，CY＋DY(H＋1)

という式で表わせます。これで車の前方に何もない時は，車を移動させます。何かあった場合は，それがハタなら1100行，チェックポイントなら1200行の処理を呼んでから車を移動させます。

メインルーチンを１回まわる間が１単位時間で，ＦＵＥＬを１消費します（620行)。これでＦＵ＝０になると燃料切れでゲームオーバーですから，1600行に飛びます。

●スペシャルフラッグ処理　(1000～1100)

1000～1040行がハタを書いたり消したりする部分で，ハタが出ている間は変数ＦＬが-1にセットされます。ハタの座標は変数ＦＸ，ＦＹにセットします。

```
590 IF OH=H GOTO 610
600 POSITIONH,XPOS(OH),YPOS(OH):ERA OH
610 MOVE H:OH=H
615 CX=CX+(H=3)-(H=1):CY=CY+(H=0)-(H=2)
620 FU=FU-1:IF FU<1GOTO1600
630 IF FU<L GOSUB1000
640 LOCATE0,21:PRINT"FUEL";FU;" "
650 GOTO 500
1000 L=L-50:IF FL GOTO1040
1010 FX=RND(26)+1:FY=RND(19)+1
1020 IFSCR$(FX,FY)<>" "THENFL=0:RETURN
1030 FL=-1:LOCATEFX,FY:PRINTCHR$(199):RETURN
1040 LOCATEFX,FY:PRINT" ":FL=0:RETURN
1100 LOCATEFX,FY:PRINT" ":SC=SC+FU/10:GOSUB1700:RE
TURN
1200 SC=SC+10:GOSUB1700
1210 IF A-48=N THEN LOCATECX+DX(H*2),CY+DY(H*2):PR
INT" ":GOTO1230
1220 LOCATECX+DX(H*2+1),CY+DY(H*2+1):PRINT" "
1230 N=N+1:IF N=10 GOTO1500
1240 RETURN
1500 ERA OH:LOCATE10,5:PRINT"FINISH!!":SC=SC+FU
1510 GOSUB1700:GOTO1620
1600 ERA OH
1610 LOCATE 5,5:PRINT"TIME OVER!"
1620 LOCATE 2,6:PRINT"YOUR SCORE IS";SC;" POINTS"
1630 IF SC<=HS GOTO1650
1640 LOCATE 5,7:PRINT"IT'S COUSE RECORD!":HS=SC
1650 LOCATE 8,9:PRINT"TRY AGAIN!!"
1660 GOTO 200
1700 LOCATE0,22:PRINT"SCORE";SC;" ":RETURN
```

1100行が車が通過した時の処理で，FUの$\frac{1}{10}$を得点に加算し，ハタを消します。

●チェックポイント通過処理　(1200〜1240)

得点に10点加算して，チェックポイントの数字を消します。最後のチェックポイント（9番）が消えたら，1500行からのゴールイン処理に飛びます。

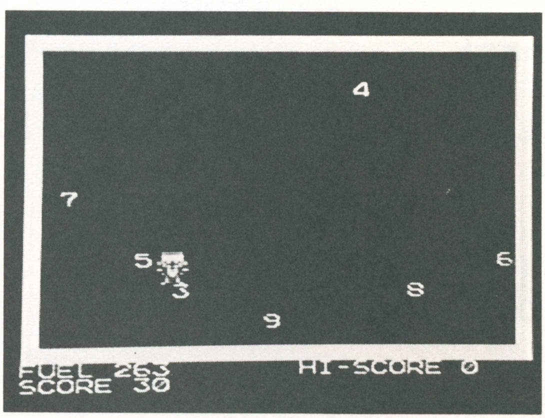
●ゴールイン，燃料切れ　(1500〜1660)

ゴールインの場合は残りのFUELを得点に加えます。

得点を表示して，ハイスコアの更新を行ない，200行に飛びます。

★変数表

CX，CY	車の位置
OH，H	車の進行方向
FU	残り燃料
FX，FY	スペシャルフラッグ位置
N	チェックポイント番号
SC	スコア
HS	ハイスコア





ファミコンのCPU

コンピューターには必ずその頭脳になるチップ（石ともいう）がある。「CPU」とか「MPU」と呼ばれるものだ。

日本の8ビットのパソコンは今までほとんどが「Z8Ø」か「68Ø9」を使っていた。PC−8001，8801，やX−1やMZシリーズが「Z8Ø」，FM−7やS−1は「68Ø9」を使っていた。

ところで，「アップル」というコンピューターを知っているかな。あの「ロードランナー」なんかが始めて走ったアメリカのコンピューターだ。この「アップル」は「65Ø2」というCPUを使っているんだ。

つまり，8ビットのCPUでは「Z8Ø」と「68Ø9」と「65Ø2」の三種類が世界の主流だったわけだ。

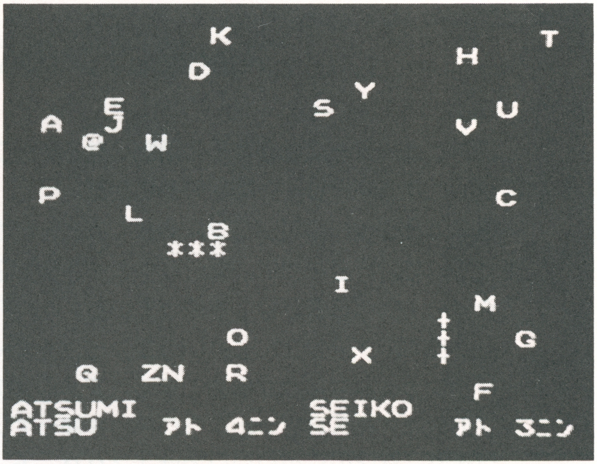
このファミコン，実は「65Ø2」のカスタム（特別製）チップを作っている。

「65Ø2」というCPU自体は，他の2つに比べると少し性能が悪いけど，良い点もいろいろあって，ゲームにはちょうどいいCPUという人もいる。何より，名機「アップル」がそれを証明しているだろう。

パソコンを持っていばってる友達に，「ぼくのはアップルと同じ石で65Ø2を使ってるんだ」と言ってみよう。でも，間違っても「ろくせんごひゃくに」なんていわないように。プログラマー仲間では「ろくごーまるに」と呼ばなければいけないんだ。ちなみに「Z8Ø」は「ぜっとはちまる」または「ぜっぱち」，「68Ø9」は「ろくはちまるく」か「まるく」と呼ぶんだ。

ガールハント





えーと今度は誰だろう。「SEIKO」か。んー，Sはどこだ。うわっ，あんな遠くか。えっちらおっちら，あ，まずい。あいつもSを狙ってる。どっちが早いか。おっといけない，間違って違う文字を拾っちゃった。@マークはどこかな…。

★ゲームの説明

このゲームは2人で遊ぶカルタ取りみたいなゲームです。2人がそれぞれ“*”と“+”印の，何やらシャクトリ虫のようなやつをコントロールして，指示された単語（名前）の文字を順番に拾い集めます。

文字は順番に，たとえば「SEIKO」ならばSから順に，E，I，…のように集めていきます。途中で間違って関係ない文字を拾ってしまったときは，@マークを探して拾ってください。この文字は，キーボードの“DEL”キーのようなもので，これを拾うとその前に拾った文字をひとつ消せます。

間違ったのをそのままにしてどんどん拾い続けてしまい，指定された名前の文字数より多くなってしまうと，せっかく拾った文字が全部（その名前の分だけです）消えてしまいます。

ひとつ名前が完成すると，次の名前が表示され，先に5人の名前を集めた方の勝ちです。

プログラムリスト

```
110 DIM CX(1),CY(1),TX(1),TY(1),H(1),M(1),N1$(1),N
2$(1),C$(1),NA$(7)
120 FORI=0TO7:READNA$(I):NEXT
130 DATA SEIKO,HIROMI,AKINA,NAOKO,YUKIKO,MOMOKO,AT
SUMI,MAIKO
140 C$(0)="*":C$(1)="+"
150 H(0)=8:N1$(0)=NA$(RND(8)):M(0)=5
160 H(1)=8:N1$(1)=NA$(RND(8)):M(1)=5
230 CLS:FORI=64TO90
240 X=RND(26)+1:Y=RND(22)
250 IFSCR$(X,Y)<>" "GOTO240
260 LOCATEX,Y:PRINTCHR$(I):NEXT
270 FORP=0TO1:GOSUB2000:GOSUB2500:NEXT
500 FORP=0TO1
510 S=STICK(P):IFS=0THENS=H(P)
515 X=CX(P):Y=CY(P)
520 IFS=1THENX=(CX(P)+1)MOD28:GOTO560
530 IFS=2THENX=(CX(P)+27)MOD28:GOTO560
540 IFS=4THENY=(CY(P)+1)MOD22:GOTO560
550 IFS=8THENY=(CY(P)+21)MOD22
560 A$=SCR$(X,Y)
570 IFA$="+"ORA$="*"GOTO750
580 CX(P)=X:CY(P)=Y:LOCATEX,Y:PRINT" ";
590 IFA$=" "GOTO 680
600 IFA$="@"THENN2$(P)=LEFT$(N2$(P),LEN(N2$(P))-1)
:GOTO620
610 N2$(P)=N2$(P)+A$
620 IFLEN(N1$(P))<LEN(N2$(P))THENN2$(P)=""
```

★遊び方

それぞれのプレイヤーは，コントローラーのⅠとⅡをそれぞれ使います。

ゲームが始まると，画面中にアルファベット26文字と，＠マークが１つの計27文字が乱数でバラまかれます。そのあとで画面の左右の端に各プレイヤーのシャクトリ虫が表示され，すぐ動き始めます。

画面の下には最初の名前が表示されます。左側がプレイヤー１(コントローラーⅠ)で，右がプレイヤー２用です。

シャクトリ虫のコントロールはコントローラーの十字ボタンで行ないます。ボタンを押した方にシャクトリ虫が進んで行き，何か文字にぶつかると，それを拾って，その文字が画面下の名前下に表示されます。

画面の上下と左右は続いていて，右に向って進んでいくと，左から出てきます。

★プログラムの説明

●初期設定　(110～270)

配列の宣言や，名前のデータ読み込み，変数の初期化を行ない，アルファベットを乱数を使ってバラまきます。130行のデータ文を変更して，たとえば名前の代わりに英語の単語にでもしておくと，遊んでいても勉強しているように見えます。

●メインループ　(500～750)

大きなFOR～NEXT文で，各プレイヤーの処理を順に行ないます。どちらのプレイヤーの処理をしているかはFOR文の制御変数であるＰで表わされます。Ｐが０のときはプレイヤー１，Ｐが１の時はプレイヤー２の処理です。

メインループは大きいので分けて説明します。

510～550行がコントローラーからの入力チェックで，移動先の座標を計算します。H(P)はシャクトリ虫の進行方向で，１が右，２左，４下，８上となっています。コントローラーの十字キーが押されていなければ前回までの進行方向にしたがって移動します。

560～600行は移動先に何があるかのチェックです。何もない（スペース）時はそのまま移動，相手のシャクトリ虫であれば停止，何かの文字なら以下の処理に進みます。

Ａ～Ｚなら，保存用の文字列Ｎ２＄(Ｐ)にその文字をつけ足します。これでＮ２＄(Ｐ)が，Ｎ１＄(Ｐ)―指示された名前が入っている―と同じになれば1000行からの処理を呼び，新しい名前をセットします。

この文字は画面から消して，すぐに新しい場所に書き直します。(650～670行)

Ｎ１＄がＮ２＄より長くなった場合はＮ２＄をクリアします。

```
630 GOSUB2000
640 IFN1$(P)=N2$(P)GOSUB1000
650 X=RND(28):Y=RND(22)
660 IFSCR$(X,Y)<>" "GOTO650
670 LOCATEX,Y:PRINTA$
680 X=TX(P):Y=TY(P)
690 IFSCR$((X+1)MOD28,Y)=C$(P)THENTX(P)=(X+1)MOD28
700 IFSCR$((X+27)MOD28,Y)=C$(P)THENTX(P)=(X+27)MOD
28
710 IFSCR$(X,(Y+21)MOD22)=C$(P)THENTY(P)=(Y+21)MOD
22
720 IFSCR$(X,(Y+1)MOD22)=C$(P)THENTY(P)=(Y+1)MOD22

730 LOCATEX,Y:PRINT" ";
740 LOCATECX(P),CY(P):PRINTC$(P);:H(P)=S
750 NEXT:GOTO500
1000 M(P)=M(P)-1:IFM(P)=0GOTO1500
1010 N2$(P)="";N1$(P)=NA$(RND(8))
1020 GOSUB2000:RETURN
1500 PLAY"C1EGDEF":CLS
1510 LOCATE7,5:PRINT"PLAYER";P+1;" WIN !!"
1520 END
2000 LOCATE-(P=1)*14,22:PRINTN1$(P)+" ";
2010 LOCATE-(P=1)*14,23:PRINTLEFT$(N2$(P)+"        "
,6);" ｱﾄ";M(P);"ﾆﾝ";
2020 RETURN
2500 CX(P)=P*27:CY(P)=10:TX(P)≒CX(P):TY(P)=CY(P)+2
2510 FORI=CY(P)TOTY(P):LOCATECX(P),I:PRINTC$(P);:N
EXT:RETURN
```

＠ならばＮ２＄の右端の１文字を消します。

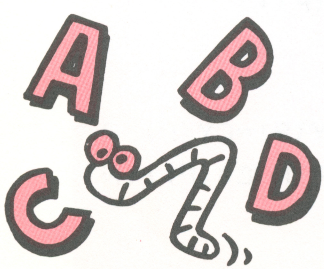
680～740行がシャクトリ虫の移動で，まず尾を消します。新しい尾の座標をしらべて(690～720)，その座標をTX(P)，TY(P)にセットします。それから前に計算しておいた新しい移動先に頭を書きます。

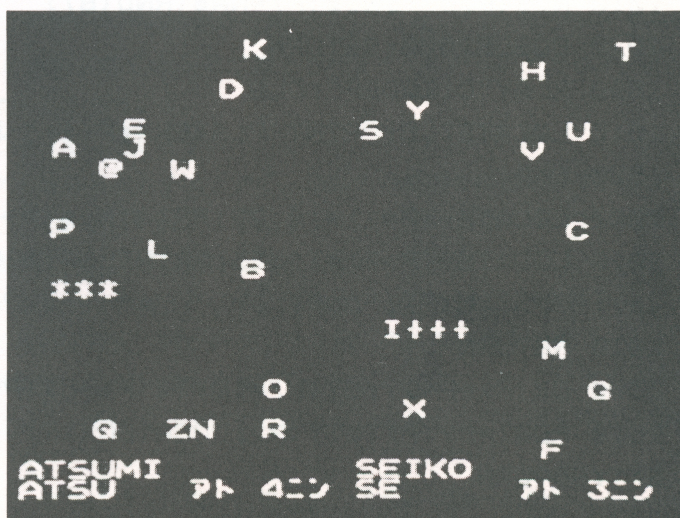
●**表示用ルーチン　(2000～2020)**

画面の下に名前の文字列，Ｎ１＄(Ｐ)とＮ２＄(Ｐ)と残りの人数を表示するルーチンです。

★変数表

P	プレイヤー識別用
CX(P)，CY(P)	シャクトリ虫の頭の座標
TX(P)，TY(P)	シャクトリ虫の尾の座標
H(P)	シャクトリ虫の進行方向
N１＄(P)	名前格納用
N２＄(P)	名前格納用
NA＄(n)	単語データ
M(P)	残り人数





コンピューターマニアは「ねくら」?

コンピューターマニアってどんなイメージかというと，圧倒的に「根暗」つまり，ねっから暗いという返事が返ってくる。

眼鏡をかけていて，ふとりぎみ(運動不足)，色は青白く（日光にあたってない)，身なりにかまわず髪はボサボサ。指にはゲームダコができている。こういう人は少々問題だね。実は，大学生や高校生にこんな人が多いんだ。

まさか君は違うだろうね。

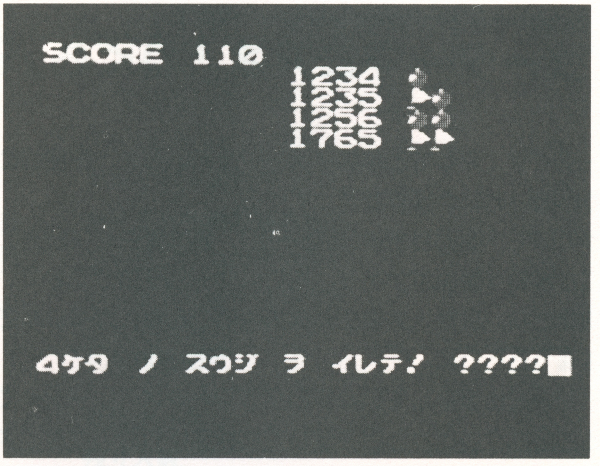
コンピューターを仕事にしているプログラマーやオペレーターでさえ，１時間に10分から20分はテレビから目を離して休まなければいけないようになってきてる。コンピューター自体が悪いんでなくて，長時間同じ姿勢でいたり，テレビを見ていると，目や体にとっても良くない事が医学的にはっきりしているんだ。

やっぱり太陽の下で思いっきりあばれて，コンピューターでも適度に遊ぶ。それが一番だ。

デバッグや，プログラム上で行きづまった時，体を動かす事が気分転換になり，かえってうまく行く時が多い。

コンピューターしか知らないなんて，やっぱり「根暗」と呼ばれてもしょうがないね。マルチ人間で，スポーツもできる，コンピューターも使えるっていうのが，やっぱし，ナウいんだよ！

MOO



１２３４。「プ，プ」 リンゴ２つか。それじゃ３１５７。「プ。」 あれ，ひとつになっちゃった。２４５７でどうかな。「プ」 やっぱりひとつだな。どうも５と７はなさそうだな。ためしに３４５７と。「プ，プ」 うん，３と４はありそうだぞ。４３７８でどうだ。「ピ，ピ」 やった，ハタ２本だ。４３はまちがいなかろう。あと使っていないのは…と，４３６９。「ピ，ピ，プ」 よしよしもう一息。４３９０かな。「ピ，ピ，ピ，ピ」「タリラリラ」，やった，ハタ４本。７回目か。80点だ。ダブルをかければよかったかな。よし次だ。

☆ゲームの説明

ＭＯＯは，ヒット　アンド　ブローとか，マスターマインドなどとも呼ばれるポピュラーなゲームですから知っている人も多いと思います。本当は人間が２人でやって競い合うものですが，相手がいない時などコンピュータ相手にやるのもいいものです。人間相手だと，たまに（よく？）ヒントを間違えたりするので，かえってコンピュータとやった方がいいのかもしれません。特に，何回やっても当てられないときのあの冷たい視線に苦しまなくていいので，練習用にちょうどいいかもしれません。

ルールを知らない人のために，このプログラムでのルールを簡単に説明しておきます。

まず，コンピュータが４ケタの数を決めます。この数字をコンピューターから返ってくるヒ

プログラムリスト

```
100 DIM N(9),S(10)
110 CLS
120 PALET B 0,47,48,39,22
130 FOR I=1 TO 10
140 READ S(I)
150 NEXT
160 DATA 1000,800,500,300,200,100,80,60,30,10
300 FOR TU=1 TO 5
310 CLS
320 FOR I=0 TO 9:N(I)=I:NEXT
330 AN$=""
340 FOR I=9 TO 6 STEP -1
350 R=RND(I+1)
360 AN$=AN$+HEX$(N(R))
370 N(R)=N(I)
380 NEXT
390 LOCATE 0,0:PRINT "SCORE";SC
500 FOR T=1 TO 10
510 LOCATE 0,15:PRINT "4ｹﾀ ﾉ ｽｳｼﾞ ｦ ｲﾚﾃ!";
520 A$="????"
530 LOCATE 17,15:PRINT A$;
540 K$=INKEY$(0)
550 IF K$=CHR$(13) THEN D=1:GOTO 620
560 IF K$=" " THEN PLAY "C1E":D=2:GOTO 620
570 IF ASC(K$)<48 OR ASC(K$)>57 GOTO 540
580 A$=RIGHT$(A$+K$,4)
590 LOCATE 17,15:PRINT A$;
600 FOR I=0 TO 300:NEXT
```

ントを使って当てるわけです。この4ケタの数は，0～9までの数を1回だけ使って決められます。つまり2234のように同じ数が重複する事はないわけです。

コンピューターの入力要求にしたがって4ケタの数を入力します。するとコンピュータは自分が決めた数と，この数を比較して，次のようなヒントを返します。

●数字とケタが合っていれば，ハタ

●数字は合っていて，ケタが違えば，リンゴ

たとえば，コンピューターが決めた数が1234で，入力した数字が1345なら，1はピッタリ合っていますが，3と4はケタがずれています。5は完全にはずれですから，結局ハタ1本（1）とリンゴ2個（3と4）がヒントとして返されます。このヒントをもとにして，さらに推理して数を当てるわけです。

ただし無制限に入力できては面白くありませんから，ひとつの問題に対して答えられるのは10回だけにします。最初は大変かもしれませんが，慣れてくるとだいたい8回以内で当てられるようになります。

あと，このプログラムでは何回目の入力で当たったかによって次のような得点がもらえます。

1回目で当たると（これを偶然という）1000点，以下，800，500，300，200，100，80，50，30，10点となっています。

さらに答えに絶対自信がある時は，“ダブル”がかけられます。これで当たれば上の得点の2倍がもらえます。ただしはずれると，それまでの得点は0になってしまいます。問題は5問だされますから，これで友人と点を競うのもいいでしょう。

★数字の入力のしかた

毎回画面やや下に

4ケタ　ノ　スウジ　ヲ　イレテ！　？？？？

と表示されます。入力は数字以外には受けつけません。1文字入力するごとに，数字がひとケタ左にずれていき，5文字以上入力すると左端の1文字が消えていきます。

正しく4ケタ入力できたら，RETURN キーを押します。この時，ダブルをかけるときはRETURN の代わりにスペースキーを押してください。

★プログラムの説明

●初期設定　(100～160)

●数字決定　(330～380)

●解答入力　(510～610)

```
610 GOTO 540
620 LOCATE 10,T:PRINT A$
630 LOCATE 0,15:PRINT "                              "
640 H=0:B=0
650 FOR I=1 TO 4
660 FOR J=1 TO 4
670 IF MID$(AN$,I,1)<>MID$(A$,J,1) GOTO 700
680 IF I=J THEN H=H+1:GOTO 700
690 B=B+1
700 NEXT:NEXT
710 HT$=""
720 IF D=2 AND H<>4 THEN SC=0
730 IF H=0 GOTO 790
740 FOR I=1 TO H
750 HT$=HT$+CHR$(199)
760 PLAY "G1"
770 NEXT
780 IF H=4 GOSUB 1000:GOTO 840
790 IF B=0 GOTO 840
800 FOR I=1 TO B
810 HT$=HT$+CHR$(215)
820 PLAY "E1"
830 NEXT
840 LOCATE 15,T:PRINT HT$
850 IF H=0 AND B=0 THEN PLAY "C3"
860 NEXT
870 LOCATE 0,0:PRINT "SCORE";SC
880 LOCATE 5,13:PRINT "ANSWER=";AN$;
890 PAUSE
900 NEXT
```

4ケタの数字の入力ルーチンです。数字は文字列にしてＡ＄にセットして返されます。

●ヒント計算　(640〜700)

4ケタの文字列をそれぞれ比較します。人間だと間違いやすい部分ですが…。ハタの数はＨ，リンゴの数はＢにセットされます。

●ヒント出力　(710〜850)

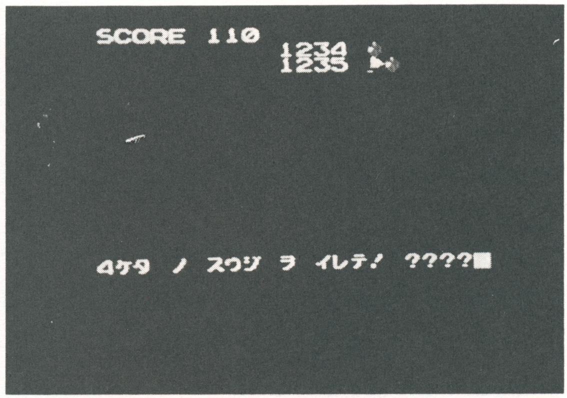
入力した数字の横に，ハタとリンゴを表示します。

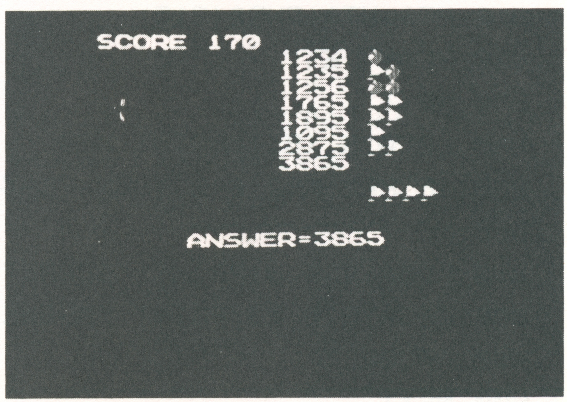
●得点，解答出力　(870〜890)

解答を出力したあと入力待ちになりますから，好きなキーを押してください。次の問題にうつります。

★変数表

AN$	コンピューターの決めた数
A$	入力した数
TU	問題番号
T	何回目の入力か
H	ハタの数
B	リンゴの数
SC	得点





```
910 END
1000 PLAY "D1ECEG"
1010 SC=SC+S(T)*D
1020 T=10:RETURN
```

入力装置のいろいろ

コンピューターが人間の命令を受け取るのにはどういう方法があるだろう。

ファミコンではキーボードを作って命令を打ち込めばいいね。ゲームをしてる時はコントローラーを使うね。このコントローラーにもいろいろある。ジョイスティックというのは，ゲームセンターのゲーム機械についてるやつと同じだ。トラックボールと言って，丸い大きな玉を手でまわすものもある。

パソコン用だと，デジタイザーやマウス（トラックボールを逆向けにしたものと思えばいい）なんてものがある。最近ワープロ用に「手書き文字認識ボード」も製品化された。ボードに字を書くと，コンピューターがそれを読み取ってくれるわけだ。

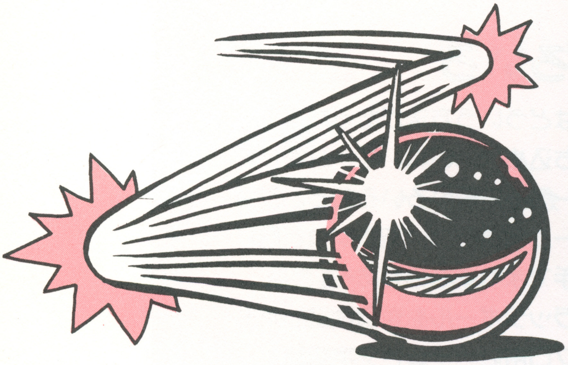
ライトペンもある。テレビの画面をこれで押すと，画面上のどこを押したかわかるもので，MSXでも使える。

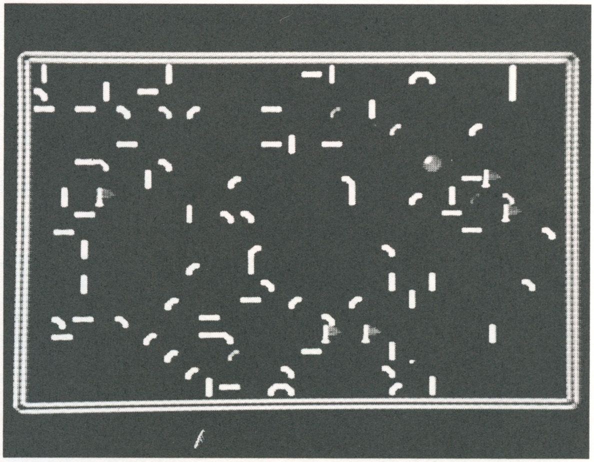
クーラーや冷蔵庫には「温度センサー」，自動ドアなら「光センサー」や「重量センサー」がついていて，温度や，光や，重さの変化をコンピューターに伝えてくれる。

コンピューターとコンピューターを結ぶには，電話を使った「音響カプラ」や，「光ファイバー」を使う。

でも何よりかっこいいのはコンピューターに話しかけるだけでいい「音声認識」法だ。これは今，日本の各メーカーが必死で開発しているけど，まだまだ高価なものだ。そういう意味では，何を話しているかは識別できないけど，声が出ているかどうかを判断できるファミコン（コントローラーIIにマイクがついていて，Ver.3ではそれを使ったゲームができる）は，時代の最先端を行くマシンとも言えるね！

REFRECTION





えーと，ここからボールを投げると，この壁にぶつかって，こっちにいってあっちにいって……。ああ，わからない。時間もないから，投げてみるか。

ゲームボードには壁がいっぱい。この中にうまくボールを投げ込んで，ハタに命中させてください。でも，この壁はボールがぶつかると，向きが変わってしまうんです。すぐ近くのハタなら当てやすいけど，壁で何回もはね返ってからハタに当たった方が，点数が高いんです。でも考えている間にも時間は経過して，あまり考えすぎると時間切れでゲームオーバー。

さあ，君はボールの動きを読み切れるか！？

★ゲームの説明

ゲームボードの周囲のレールの上を，ボールを動かして好きな所で発射してください。ボールは，そこら中で壁にぶつかりながら飛び回ります。そのうちにうまくハタに当たれば，壁に当たった回数×10点が加算されます。ハタを全部消すと一面クリアで，壁の数が少し増えて次の面に進みます。壁の数は１面が100個で，20個ずつ増えていき，７面で100個にもどります。１面クリアごとに，残り時間がボーナスとして得点になります。

プログラムリスト　（BG画面データーは103ページ）

（BG画面データーは103ページ）

```
100 DIM W$(3)
104 CGSET1
105 PALET B 0,47,48,33,48
106 PALET B 1,47,48,48,48
107 PALET B 2,47,22,48,48
110 CLS:CGEN3:VIEW:SPRITEON
140 FORI=0TO3:READ S:W$(I)=CHR$(S):NEXT
150 DATA 237,239,238,240
160 DEFSPRITE0,(2,0,0,0,0)=CHR$(207)
300 WL=0:SC=0:LOCATE5,13:PRINT"PUSH START BUTTON!"
310 IFSTRIG(0)<>1GOTO310
320 VIEW:TM=600:SX=14:SY=20:F=5
330 FORI=1TO F
340 X=RND(26)+1:Y=RND(19)+1
350 IFSCR$(X,Y)<>" "GOTO340
355 COLORX,Y,2
360 LOCATE X,Y:PRINTCHR$(199):NEXT
362 FORI=1TO100+WL
363 X=RND(26)+1:Y=RND(19)+1
364 IFSCR$(X,Y)<>" "GOTO363
365 LOCATE X,Y:PRINTW$(RND(4)):NEXT
370 GOSUB2100:GOSUB2150
380 GOSUB2200:LOCATE0,21:PRINT"TIME";TM/10;" "
500 IFSTRIG(0)=8GOTO1000
510 S=STICK(0)
520 X=SX+(S=2)-(S=1):Y=SY+(S=8)-(S=4)
550 IF X<0 OR X>27 OR Y<0 GOTO 590
```

★壁の種類

①┃　②╮　③━　④╭

の４種類で，╮と╭はそれぞれ＼と／のつもりです。②と④はボールの進行方向を90°変えます。

①と③は，●──→▯ のようにぶつかった時はボールの進行方向を180°変えます（来た方向に戻る）。

ボールが，●──→▭ の向きにぶつかった場合は，ボールはそのまま直進します。

ボールが壁にぶつかると，壁は必ず左（反時計まわり）に45°回転します。つまり①は②に，②は③に，③は④に，④は①になるわけです。

★遊び方

プログラムを走らせると，

PUSH　START　BUTTON！

と表示されますから，コントローラーⅠのスタートボタンを押してください。ゲームが始まると，画面左下に残り時間と得点が表示されます。

コントローラーの十字ボタンでボールの発射位置を決め，Ａボタンで発射します。ボールが発射され，ゲームボードの中を動きまわっている間は時間は減りません。

★プログラムの説明

●ボールの発射位置を決める　（500〜600）

コントローラーからのキー入力を調べ，それにしたがってボールを移動させます。Ａボタンが押されたら，1000行に飛びます。

●ボール発射　（1000〜1050）

ボールの進行方向（０〜３）を決め，変数Ｈにセットします。

●ボール移動と衝突　（1060〜1220）

ボールを進行方向に１ブロック進め，何にぶつかったかを調べます。ぶつかった物のコードを変数Ｉにセットして各処理を行ないます。

コードは次のとおりです。

32：何もない

199：ハタ

```
560 I=ASC(SCR$(X,Y))
570 IFI<222OR I>227GOTO590
580 SX=X:SY=Y:GOSUB2200
590 TM=TM-1:IFTM<1THENVIEW:GOTO1750
600 LOCATE0,21:PRINT"TIME";TM/10;" ":GOTO500
1000 IFI<226GOTO590
1010 IFSX=0THENH=1:GOTO1050
1020 IFSX=27THENH=3:GOTO1050
1030 IFSY=0THENH=2:GOTO1050
1040 IFSY=20THENH=0:GOTO1050
1050 X=SX:Y=SY:C=0
1060 SX=SX+(H=3)-(H=1):SY=SY+(H=0)-(H=2):GOSUB2200
1080 I=ASC(SCR$(SX,SY))
1090 IFI=32GOTO1060
1095 C=C+1
1100 IFI=199GOTO1600
1110 IFI=226ORI=227GOTO1660
1120 ONI-236GOTO1130,1160,1190,1210
1130 K=0:IF H MOD2=0THENH=(H+2)MOD4:GOTO1500
1140 GOTO1500
1160 K=2:IF H MOD2=1THENH=(H+2)MOD4:GOTO1500
1170 GOTO1500
1190 K=1:IF H MOD2=0THENH=H+1:GOTO1500
1200 H=H-1:GOTO1500
1210 K=3:IF H MOD2=0THEN H=(H+3)MOD4:GOTO1500
1220 H=(H+1)MOD4:GOTO1500
1500 PLAY"C0":K=(K+1)MOD4
1510 LOCATE SX,SY:PRINTW$(K):GOTO1060
1600 SC=SC+C*10:PLAY"C0DE":F=F-1
1650 LOCATE SX,SY:PRINT" ":GOSUB2100:IFF=0GOTO1700
1660 SX=X:SY=Y:GOSUB2200:GOTO500
```

226，227：周囲のレール

237～240：壁

壁にぶつかった時の処理は1120～1220行で，ここでボールの新しい進行方向を決め，ぶつかった回数（変数C）を１増やし，壁を回転させます（1500行）。

以上が中心となる部分で，この他には，

ハタクリア（1600），１面クリア（1700），ゲームオーバー（1750），ボール表示（2200）などがあります。

初期設定は，100～380行で，ハタのセットは330～360，壁のセットは362～365行で，それぞれ表示位置は乱数で決めています。

BG画面のデーターを変えると，ゲームボードの形を好きな形にすることができますが，この変更をする時は，ハタと壁の座標に気をつけてください。

ハタの数を変えたい人は，320行のF＝5を，制限時間を変えたい人は，320行のTM＝600を好きな値に変えてください。

壁の数を変えたい人は362行の，

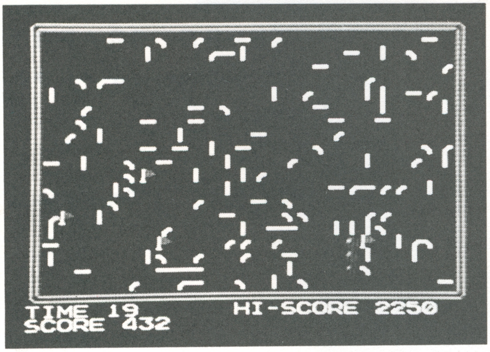
```
FOR I = 1 TO100 + WL
```

の100を好きな値にしてください。多い方が見てて楽しくなりますが，動きの予想は難しくなります。

★変数表

SX，SY	ボールの位置
H	ボールの進行方向
C	ボールが壁にぶつかった回数
F	ハタの数
TM	残り時間
PX，PY	ボールのスタート位置
SC	スコア
HS	ハイスコア

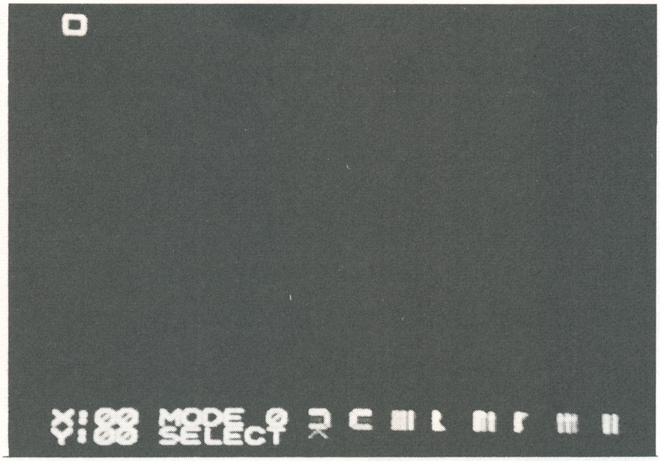
```
1700 VIEW:LOCATE5,5:PRINT"MISSION COMPLETED!!"
1720 SC=SC+100+TM/10:GOSUB2100
1730 WL=(WL+20)MOD100:FORI=0TO999:NEXT:GOTO320
1750 LOCATE10,7:PRINT"GAME OVER"
1760 LOCATE2,8:PRINT"YOUR SCORE IS";SC;" POINTS"
1770 IFSC<=HS GOTO1800
1780 LOCATE5,9:PRINT"IT'S HI-SCORE !!":HS=SC:GOSUB
2150
1800 LOCATE10,11:PRINT"TRY AGAIN!":GOTO300
2100 LOCATE0,22:PRINT"SCORE";SC:RETURN
2150 LOCATE13,21:PRINT"HI-SCORE";HS:RETURN
2200 SPRITE0,SX*8+16,SY*8+24:RETURN
```





BG画面テクニック

BGグラフィックの入力法



●BG画面書き込みモード

Ver.2では，GAME　BASICにしたあと，「2－BG　GRAPHIC」を選んでください。すでに，BASIC画面にしている場合はダイレクトで，

　SYSTEM

とすると，選択モードになるので，「2」のキーを押します。（プログラムは消えないので安心してください）

Ver.3では，電源ONのあと，ダイレクトで，

　BGTOOL

としてください。

BG画面の書き込みモードになります。**（写真）**

カーソルキーでカーソルを移動します。カーソルの座標は左下にX，Y座標で表示されています。リターンキーを押すとカラーが変ります（Ø→1→2→3→Ø)。右下に8つならんだキャラクターをキャラクターセットと言い，これは図－1の横一列にあたります（開始画面ではA列が表示)。CLR・HOMEキーを押すと，A→B→C…→M　と表示列が変わります。SHIFTキーとCLR・HOMEキーをいっしょに押すと，M→L→K…→A　と逆に変ります。

●図１
ＢＧグラフィックキャラクター



「ＤＥＬ」キーを押すとセレクトカーソルが右へ，「ＩＮＳ」キーを押すと左へ移動します。

以上のキーを使い目的の場所と色とキャラクターを選びスペースキーを押すと，そのキャラクターがＳＥＴされます。

間違えたキャラクターを消すには，カーソルキーをそのキャラクターの上に持っていき文字キーの「Ｄ」を押すと消えます。

ＢＧ画面データは「Ｅ５Ø」のように３文字でになっています。これは，キ

ャラクター表（図－１）の「E」列のナンバー「５」（左から６番目）のキャラクターで，カラーは「∅」という意味です。

ＢＧグラフィックの書き込みが終るとテープへＳＡＶＥします。ＥＳＣキーを押すと画面左上にいくつもの文字が出てきて，その左に「＞」マークが点滅します。カーソルキーでこの「＞」マークを「ＦＩＬＥ」のところへあわせてスペースキーを押します。

ＬＯＡＤかＳＡＶＥか聞いてきますから「Ｓ」キーで押すとファイルネームを聞いてきます。１６文字以内でファイルネームを入力後ＲＥＴＵＲＮキーを押すとＳＡＶＥが始まります。

このＢＧ画面のデーターのあとにプログラムをＳＡＶＥすると，ＬＯＡＤの時便利になります。

Ver.3では，ＢＡＳＩＣ画面上からＢＧ画面を呼び出さなくても直接にＢＧ画面データをＬＯＡＤ，ＳＡＶＥできるようになりました。

　　ＬＯＡＤＳ　“ファイル名”

　　ＳＡＶＥＳ　“ファイル名”

という命令を使います。

また，ダイレクトで

　　ＢＧＧＥＴ

と入力するとＢＧ画面のデーターがプログラム領域に転送されるので，このあと，

　　ＳＡＶＥ　“ファイル名”

とすると，プログラムとＢＧ画面データーを一度にテープにＳＡＶＥできます。

このテープをＬＯＡＤするには，

　　ＬＯＡＤ　“ファイル名”

　　ＢＧＰＵＴ

とします。この「ＢＧＰＵＴ」は，プログラム上にあるＢＧ画面データをＢＧ画面に転送する命令です。

なお，この方法はプログラムがあまり長すぎると，

　　？ＯＭ　ＥＲＲＯＲ

になります（この本に入っているプログラムは大丈夫です。）

このＢＧ画面の入力には，「ＣＯＰＹ」や「ＭＯＶＥ」などの便利な入力法があります。「ファミリー・ベーシック・ホップ編」にくわしい操作法が説明されているので，参考にしてください。

FLYFLY BG

	Ø					5					1Ø					15					2Ø					25		27
Ø																			I6Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	I7Ø
										G62			G52		G52				J2Ø	H	I	–	S	C	O	R	E	J2Ø
			G52					G62											J2Ø									J2Ø
						G62				G52		G52				G52			J2Ø									J2Ø
		G52																	J2Ø		S	C	O	R	E			J2Ø
5												G52	G52						J2Ø									J2Ø
							G52		G62				G52						J2Ø									J2Ø
			G52							G62							G52		J2Ø									J2Ø
												G62					G52		J2Ø									J2Ø
										G52									J2Ø		F	U	E	L				J2Ø
1Ø																			J2Ø									J2Ø
		G62																	J2Ø									J2Ø
					G52										G52				J2Ø									J2Ø
		G62											G52		G62		G62		J2Ø		B	O	N	U	S			J2Ø
					G62					G52									J2Ø									J2Ø
15	G62			G62										G52	G52				J2Ø									J2Ø
								G62									G52		J2Ø									J2Ø
			G52																J2Ø		S	H	I	P				J2Ø
					G62					G62									J2Ø									J2Ø
					G62								G62						J2Ø									J2Ø
2Ø																			JØØ	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J1Ø

アステロイド BG

	Ø					5					1Ø					15					2Ø					25		27
Ø	G32																					G22	G22	G22	G22	G22	G22	G22
	G32						G62							G52					G62			G22	G22	G22	G22	G22	G22	G22
	G32		G52									G62			G52							G22	G22	G22	G22	G22	G22	G22
	G32								G52							G62		G52				G22	G22	G22	G22	G22	G22	G22
	G32																G62					G22	G22	G22	G22	G22	G22	G22
5	G32													G52					G62			G22	G22	G22	G22	G22	G22	G22
	G32			G52			G62			G62			G62			G62							G22	G22	G22	G22	G22	G22
	G32																			G52				G22	G22	G22	G22	G22
	G32	M6Ø	M6Ø	M6Ø	G12													GØ2						G22	G22	G22	G22	G22
	G32	X		2	G32												GØ2	G42	G12			G62		G22	G22	G22	G22	G22
1Ø	G32	G32	G32	G32	G32												G22	G42	G22					G22	G22	G42	G42	G42
	G32	G32	G32	G32	G32							M6Ø	M6Ø	M6Ø	G12		G22	G42	G22		G62			G22	G22	G42	G42	G42
	G32	G32	G32	G32	G32		G62	G52				X		2	G32	G42	G42	G42	G22		G52			G22	G22	G42	G42	G42
	G32	G32	G32	G32	G32							G22	G22	G32	G32	G42	G42	G42	G42						G22	G42	G42	G42
	G32	G32	G32	G32	G32							G22	G22	G32	G32	G42	G42	G42	G42				G52		G22	G42	G42	G42
15	G32	G32	G32	G32	G32	G12					GØ2	G22	G22	G32	G32	G42	G42	G42				G62			G22	G42	G42	G42
	G42	G42	G42	G42	G42	G42	M5Ø	M5Ø	M5Ø	M5Ø	G42	G42	G42	G42	G42	G42	G42	G42							G22	G42	G42	G42
	G42	G42	G42	G42	G42	G42	X		1	G42	G42	G42	G42	G42	G42	G42	G42	G42						GØ2	G22	G42	G42	G42
	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	M7Ø	M7Ø	M7Ø	M7Ø	G42	G42	G42	G42	G42	G42
	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	X		5	G42	G42	G42	G42	G42	G42
2Ø	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42

FLYFALL BG

	0					5					10					15					20					25		27
0																												
					G62																							
										G52				G52					G62										
					G52																		G52					
																					G52							
5										G52			G62															
			G62																		G62							
							G52					G62													G62			
															G52													
					G52																							
10																		G62										
									G52														G52					
					G62								G62			G52												
15																								G52				
				G52							G62			G52							G62								
																		G62										
20	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42	G42

コマンダー BG

	Ø					5					1Ø					15					2Ø					25		27
Ø	*	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	*	*	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	*	I6Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	I7Ø
	D4Ø																			D4Ø	J2Ø							J2Ø
	D4Ø																			D4Ø	J2Ø							J2Ø
	D4Ø			G72	G72			G72	G72			G72	G72			G72	G72			D4Ø	J2Ø	S	C	O	R	E		J2Ø
	D4Ø			G72	G72			G72	G72			G72	G72			G72	G72			D4Ø	J2Ø							J2Ø
5	D4Ø																			D4Ø	J2Ø							J2Ø
	D4Ø																			D4Ø	J2Ø							J2Ø
	D4Ø			G72	G72			G72	G72			G72	G72			G72	G72			D4Ø	J2Ø							J2Ø
	D4Ø			G72	G72			G72	G72			G72	G72			G72	G72			D4Ø	J2Ø							J2Ø
	*																			*	J2Ø							J2Ø
1Ø	*																			*	J2Ø	S	H	I	P			J2Ø
	D4Ø			G72	G72			G72	G72			G72	G72			G72	G72			D4Ø	J2Ø							J2Ø
	D4Ø			G72	G72			G72	G72			G72	G72			G72	G72			D4Ø	J2Ø							J2Ø
	D4Ø																			D4Ø	J2Ø							J2Ø
	D4Ø																			D4Ø	J2Ø							J2Ø
15	D4Ø			G72	G72			G72	G72			G72	G72			G72	G72			D4Ø	J2Ø							J2Ø
	D4Ø			G72	G72			G72	G72			G72	G72			G72	G72			D4Ø	J2Ø							J2Ø
	D4Ø																			D4Ø	J2Ø							J2Ø
	D4Ø																			D4Ø	J2Ø							J2Ø
	*	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	*	*	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	D4Ø	*	JØØ	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J1Ø
2Ø																												

カメレース BG

	Ø					5					1Ø					15					2Ø					25		27
Ø				K72	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K22	K52	LØ2
				K62																						K62		K62
				K62																						K62		K62
				K42	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	KØ2	K52	K32
				K62																						K62		K62
5				K62																						K62		K62
				K42	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	KØ2	K52	K32
				K62																						K62		K62
				K62																						K62		K62
				K42	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	KØ2	K52	K32
1Ø				K62																						K62		K62
				K62																						K62		K62
				K42	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	KØ2	K52	K32
				K62																						K62		K62
				K62																						K62		K62
15				K42	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	KØ2	K52	K32
				K62																						K62		K62
				K62																						K62		K62
				L12	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K12	K52	L22
2Ø																												

チックタックBOMB BG

	Ø					5					1Ø					15					2Ø					25		27
Ø		M6Ø	M6Ø	M6Ø	M6Ø	H	I	—	S	C	O	R	E	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø			
5		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
1Ø		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
15		M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø	M6Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M7Ø	M6Ø			
		M6Ø	M6Ø	M6Ø	M6Ø	B	O	N	U	S	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø	M6Ø			
2Ø	S	C	O	R	E									P	E	N	P	E	N									

MARIO KICK BG

	0					5					10					15					20					25		27
0	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	I60	J30	J30	J30	J30	J30	J30	J30	J30	I70
	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	J20	H	I	—	S	C	O	R	E	J20
	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	J20									J20
	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	J20									J20
	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	J20									J20
5	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	J20		S	C	O	R	E			J20
	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	J20									J20
	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	J20									J20
	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	J20									J20
	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
10	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
15	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20		M	A	R	I	O			J20
	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	M53	J20									J20
																			J20									J20
20																			J00	J30	J30	J30	J30	J30	J30	J30	J30	J10

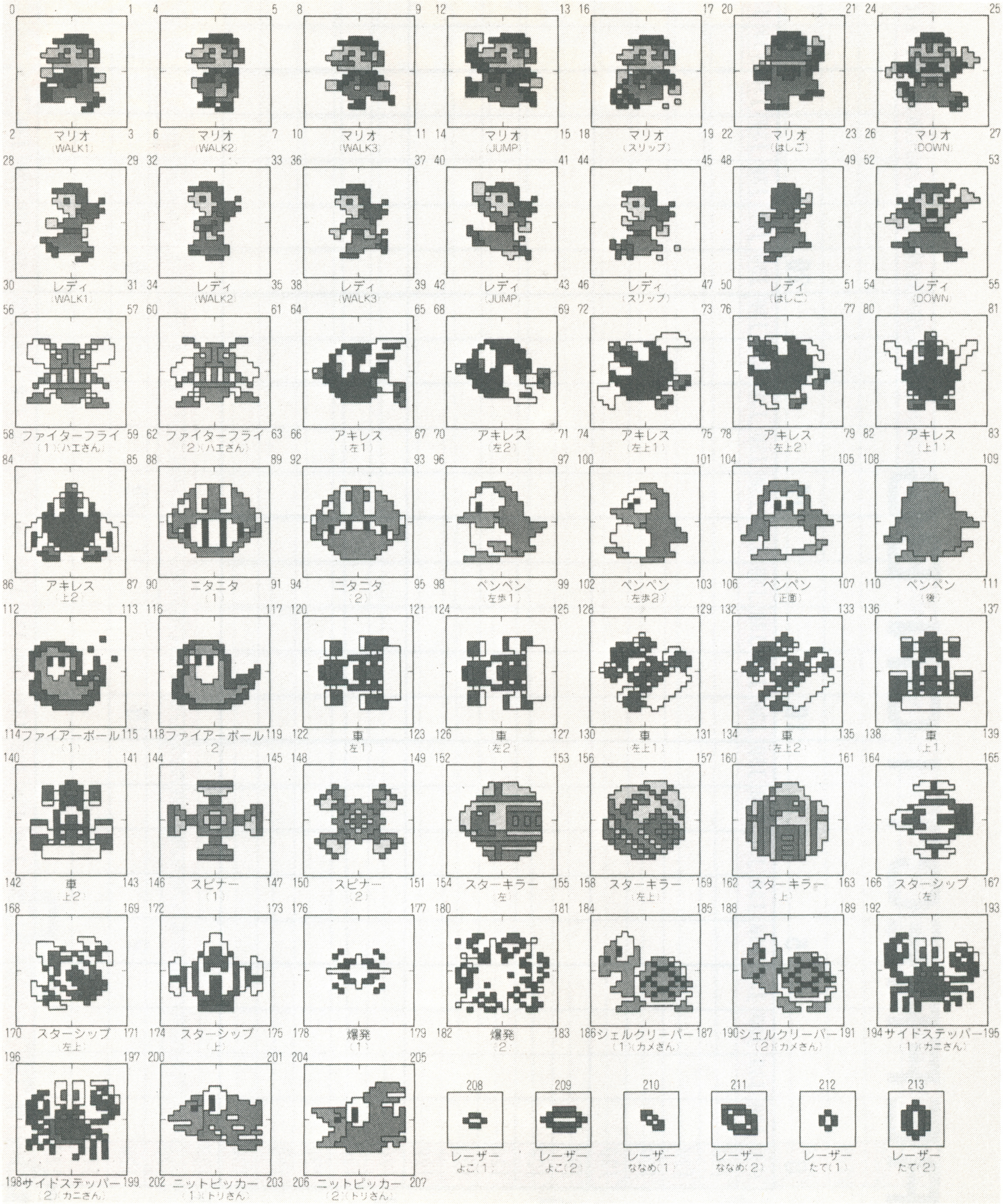
ラリー BG

	Ø					5					1Ø					15					2Ø					25		27
Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
5	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
1Ø	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
15	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
	M5Ø																											M5Ø
2Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø	M5Ø

REFRECTION BG

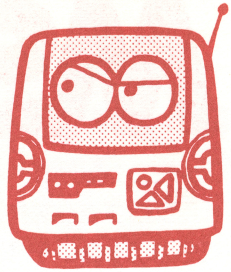
	Ø					5					1Ø					15					2Ø					25		27
Ø	I6Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	I7Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
5	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
1Ø	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
15	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
	J2Ø																										J2Ø
2Ø	JØØ	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J3Ø	J1Ø

ゲームキャラクター表



0 1 2 マリオ (WALK1) 3
4 5 6 マリオ (WALK2) 7
8 9 10 マリオ (WALK3) 11
12 13 14 マリオ (JUMP) 15
16 17 18 マリオ (スリップ) 19
20 21 22 マリオ (はしご) 23
24 25 26 マリオ (DOWN) 27
28 29 30 レディ (WALK1) 31
32 33 34 レディ (WALK2) 35
36 37 38 レディ (WALK3) 39
40 41 42 レディ (JUMP) 43
44 45 46 レディ (スリップ) 47
48 49 50 レディ (はしご) 51
52 53 54 レディ (DOWN) 55
56 57 58 ファイターフライ (1)(ハエさん) 59
60 61 62 ファイターフライ (2)(ハエさん) 63
64 65 66 アキレス (左1) 67
68 69 70 アキレス (左2) 71
72 73 74 アキレス (左上1) 75
76 77 78 アキレス (左上2) 79
80 81 82 アキレス (上1) 83
84 85 86 アキレス (上2) 87
88 89 90 ニタニタ (1) 91
92 93 94 ニタニタ (2) 95
96 97 98 ペンペン (左歩1) 99
100 101 102 ペンペン (左歩2) 103
104 105 106 ペンペン (正面) 107
108 109 110 ペンペン (後) 111
112 113 114 ファイアーボール (1) 115
116 117 118 ファイアーボール (2) 119
120 121 122 車 (左1) 123
124 125 126 車 (左2) 127
128 129 130 車 (左上1) 131
132 133 134 車 (左上2) 135
136 137 138 車 (上1) 139
140 141 142 車 (上2) 143
144 145 146 スピナー (1) 147
148 149 150 スピナー (2) 151
152 153 154 スターキラー (左) 155
156 157 158 スターキラー (左上) 159
160 161 162 スターキラー (上) 163
164 165 166 スターシップ (左) 167
168 169 170 スターシップ (左上) 171
172 173 174 スターシップ (上) 175
176 177 178 爆発 (1) 179
180 181 182 爆発 (2) 183
184 185 186 シェルクリーパー (1)(カメさん) 187
188 189 190 シェルクリーパー (2)(カメさん) 191
192 193 194 サイドステッパー (1)(カニさん) 195
196 197 198 サイドステッパー (2)(カニさん) 199
200 201 202 ニットピッカー (1)(トリさん) 203
204 205 206 ニットピッカー (2)(トリさん) 207
208 レーザー よこ(1)
209 レーザー よこ(2)
210 レーザー ななめ(1)
211 レーザー ななめ(2)
212 レーザー たて(1)
213 レーザー たて(2)

プログラムが動かない時は？



プログラムをちゃんと打ち込んだはずなのに動作がおかしい，エラー・メッセージが出て止ってしまうことがあります。これは，プログラムの間違いでない限り入力ミスです。

SN，TM，各エラーは入力のタイプミスです。その行に打ち間違いがないか確かめてください。（ILエラーの一部も）

NF，RG，OD，UL，DD，MO，UF(Ver.3）のエラーは，入力しなければいけない命令や行やオペランド（LOCATE　0，0のような命令の「0，0」にあたる部分）を打ち忘れているか，逆に同じ命令をだぶって2回入力している時に起こります。

IL，OV，OM，SD，DZエラーは，どこかで変数の値が予定外になっている事が多く，「?OV　ERROR　IN　250」と出ても，その原因は，その行以前にある場合が多いのです。たとえば，

```
10  A=100
20  B=A*A
```

というプログラムの10行を「A＝1000」と打ち間違うと，エラーは20行で発生します。

エラーは出ないけどおかしな動きをする場合も，どこかで変数の値がおかしくなっている（偶然エラーの出ない程度の）ことが多いようです。

エラーを出した変数，おかしな動きをする原因の変数をつきとめて，その変数が出てくる行を全部確かめてください。Ver.3では，プログラム中の文字列を探す命令があるので便利です。

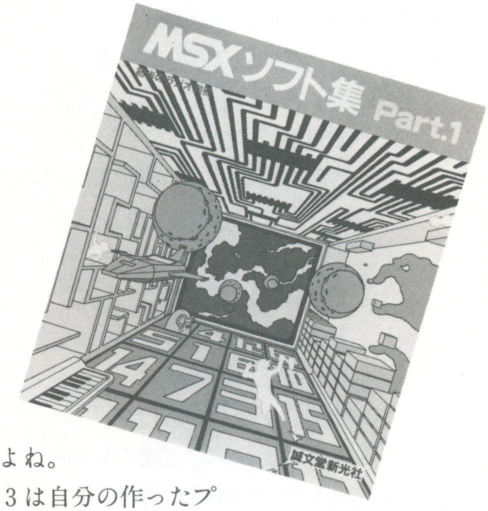
```
FIND  "A"
```

とするとプログラム中で「A」という文字がある行を全部表示してくれます。

このように，プログラムの誤りや入力ミスを「バグ」と呼び，それを訂正することを「デバッグ」と呼びます。「ファミリー・ベーシック・ジャンプ編」ではこの「デバッグ」の方法をくわしく説明しています。

MSXソフト集

ファミコンだけが，ゲーム機じゃない!!　BASICが走るのも，ファミコンだけじゃない!!　プログラムを作る楽しさ，自分で作るゲームで遊ぶ楽しさは，ファミコンの数倍なんてものじゃないぞ!!　でも，BASICでどのようにプログラムしたらいいのか分からない君に，初歩のラジオの筆者の高木春花をはじめNATSのメンバーが，たくさんのプログラムを提供しますよ!!
では，MSXソフト集のパート1から，ソフトの内容を紹介しましょう。



Part.1

には，ソフト10本を掲載しています。

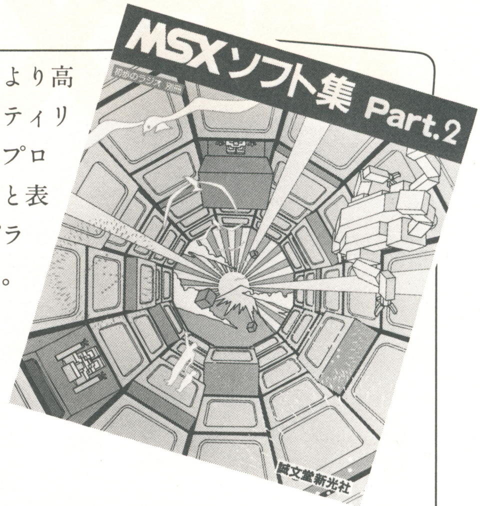
1．ブロックくずし　2．ホームデータベース
3．スーパーバックアップ　4．コンピュレコーダー
5．スーパーメイズ　6．ハイパーチェイサー
7．カセットファイルズ　8．バラン　9．15ゲーム
10．もぐらたたき　です。
1，9，10は，TVゲームなどで君達もよく知っているよね。
2，3，4，7は，実用ソフトとも言える物なんだけど，3は自分の作ったプログラムや友達のソフトを借りたとき，1本のテープに何本ものプログラムが記録してあるときなんかに簡単にコピーしちゃおうというもの。4は，MSXのCMTのLOAD端子から音をメモリーに記録して再生しちゃうというものなどなど，盛りだくさん。

Part.2

では，プログラムを7本掲載しています。

1．たまえの文字当てゲーム　2．フライング・スポット　3．スローコマンド　4．スプライトヘルパー　5．Combot 199 X　6．MSXでアニメーション
ゲームは1，5で，SCREEN　1のテキストモードでグラフィックをやっちゃおうという

もの。なぜか，グラフィックモードで作ったゲームより高速ですよ。２はリアルタイムCGをするためのユーティリティソフト。３はプリンターを持っていない人が，プログラムのデバッグをするときに，LISTをゆっくりと表示するためのソフト。４はMSXの弱点であるスプライトを，４つ以上表示しちゃうというスゴイソフト。６では，MSXでアニメーションを作るにはどうしたらよいのかを，サンプルプログラムと解説をしています。



Part.3

では，ソフト５本と少ないですが，中身の濃いソフトばかり。

1. Wolf BOY　2. SCEIN—EDIT
3. リモートボンバー　4. コンピュ・キーヤー
5. MSXアニメーション２

ゲームはこのなかの１と３の２本。２はパート２のCombotと今回のWolf BOY，リモートボンバーの画面を作るためのエディターで，４は，アマチュア無線とMSXのドッキング。自分の押したキーがすぐにモールス符号になっちゃうというソフト。５では前回に引き続きMSXでアニメーションをやらせるためのテクニック大公開!!



Part.4

では，

プログラムを作ったけど，もう少し早くならないかな，なんて思っている方にテクニック公開や，いまはやりのRPG（ロールプレイングゲーム）を作っちゃったぞ。などなどプログラムはもりだくさん。５月下旬に発売予定

検印省略 NDC 548

ファミコンソフト集 001

昭和60年5月27日 発 行

定価 800円

著 者 地球防衛軍

発行者 小川茂男

発行所 ㈱誠文堂新光社
東京都千代田区神田錦町1—5—5
郵便番号 101
電話 東京 (292) 1211
振替口座 東京 7—6294
印刷・広研印刷株式会社
製本・岡嶋製本工業

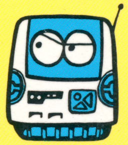
© 地球防衛軍
Printed in Japan
(本誌掲載記事の無断転用を禁じます)
万一落丁・乱丁の場合はお取替えいたします

ISBN4-416-18515-4 C2055

本社発行の雑誌 子供の科学/天文ガイド/MJ無線と実験/初歩のラジオ/農耕と園芸
商店界/アイデア/ブレーン/ザ・コピーライターズ/月刊 芽
ガーデンライフ/愛犬の友/フローリスト/囲碁/DEVICE file

ファミコン大作戦

ファミコンソフト集 001



ファミコン大作戦

誠文堂新光社

ISBN4-416-18515-4 C-2055 ¥800E 定価 800 円

●ファミリーコンピュータは任天堂の商標です。

ファミコン大作戦 ファミコンソフト集 001

誠文堂新光社

ファミコン大作戦

ファミコンソフト集 001

誠文堂新光社

郵便はがき

料金受取人払

101

神田局承認

2001

差出有効期間
昭和60年7月
7日まで

切手を貼らずに
お出しください

（受取人）
東京都千代田区神田局区内
神田錦町1の5の5

株式会社
誠文堂新光社

『ファミコン大作戦』
編集部 行

読者アンケート

ご住所

郵便番号

お名前（ふりがな）

ご年齢

●ゲームソフトは何本持っていますか
（　　　　　　本）

●市販のソフトの価格は高いですか，安いですか
（　高い　　安い　）

●いくらぐらいが適当だと思いますか
（　　　　　　円）

●あなたの欲しいパソコンは何ですか
（　　　　　　）

ゲームソフト 001 発売予定

「ファミコンソフト集001」に掲載のゲームソフトを一本のテープに，すべて集録しました。

予定価 検討中

●予約御希望の方は○印をおつけください。

テープを予約　する
　　　　　　　しない

☆ このハガキのアンケートに答えていただいた方々の中から抽選で20名様に発売予定のゲームソフト（カセットテープ）をプレゼントします。

●申し込み締切り日　　昭和60年7月7日（消印有効）

ファミコンについて，わからないこと，知りたいことなどをお書きください。

誠文堂新光社
新刊案内　1985・5



■ご注文は、なるべくお近くの書店にお願いいたします。
■直接小社にご注文の場合は、著者名・書名・冊数および住所氏名を明記のうえ、代金に送料３００円を加えてお送りください。
■現品がお手元に届くまで時間がかかることがありますので、あらかじめご了承ください。
■ご送金の際は、「振替口座東京7-6294番」、または現金書留をご利用ください。定価は、'85年４月現在のものです。
■詳細ご希望の方は、図書目録(84.8)をご請求ください。

若者のためのデザイン総合誌No.２　５月10日発売！

PORTFOLIO

<No.２のおもな内容>
●楽しいジュエリーの展覧会●R.コペッキーのユーモラスなイラストレーション●デジタル・アーキテクチュアは可能か●村上基浩氏の製作プロセス公開●J.フォリス"石"のコレクション●スタジオ紹介
奇数月10日発売・A4変・96ページ・1000円

マッハ４で駆ける音速コピー雑誌No.４　好評発売中！

ザ・コピーライターズ

<No.４のおもな内容>
●大イラストレーション航海時代●チェッカーズ現象の秘密●楽しい広告のお勉強●糸井重里の500行●What'sナイトタイムス●コピーライターへの道●サントリー変調●松田聖子の大変身●人生相談ほか
1・4・7・10月の毎１日発売・A4変・128ページ・定価680円

マイコン＆インターフェイスの技術情報誌No.２　好評発売中！

DEVICE file デバイスファイル

<No.２のおもな内容>
●特集・コンピュータのデータ通信●アセンブラ、BASICプログラマのためのＣ言語入門●各社最新デバイス・ファイル●DMA転送方式高速A/Dコンバータ●DR社　コンカレントDOS286を発表
1・4・7・10月の毎18日発売・B5・136ページ・定価650円

誠文堂新光社　〒101 東京都千代田区神田錦町１−５
☎ 03−292−1211(代表)・振替東京７−6294

今月の新刊

ニューヨークのパッケージデザイン
八尾武郎編
定価12000円

ニューヨークを中心に第一線で活躍するパッケージ・プロダクション15社の作品と全貌を紹介。

パソコンによるシステム設計
戸内順一著
定価1800円

パソコン本体+周辺機器でシステムを組む場合の実務上の手法を、リストをあげ、具体的に解説。

よくわかるBASIC&プログラミング入門
PC—9801E/F/M
湯田幸八著
定価2200円

PC9801E/F/Mに対応するBASICとプログラミングを完全マスターできる入門書。

Logoコマンドブック
GEODESIC編
定価2000円

現在使用されている、すべてのLogoのコマンド対応と、それぞれの特徴をわかりやすく解説。

パソコンは3日で打てる！
日常語で打つ、カタコトBASIC
●BASICビジネス編
山本嘉虎著
定価800円

専門用語をすべて一般語で理解していくドッグ・ベース方式を使って解説するユニークな入門書。

はじめて見るマイコンの本
奥澤清吉著
定価1200円

マイコンの内部構造から操作法などまで解説する、小学校高学年の人から楽しめるマイコン書。

海賊列伝
小島敦夫著
定価2000円

本書は海賊の実像をヨーロッパ古代・中世に求めて、人物を中心にその活動を描き出します。

図解りんごのわい化栽培
小池洋男著
予定価2600円

りんごのわい化栽培のノウハウを各地の事例をあげながら、図解中心に解説する実際入門書。

野菜と果実のビタミンC入門
斎藤進著
定価980円

生産・流通から消費者の手にわたるまでのビタミンCの実態を栄養学の立場から詳しく解説。

新編ヨセ辞典
加納嘉徳著
定価2000円

49年に刊行されたヨセ辞典の内容を一新。ヨセのテクニックのすべてを網羅した決定版。

'84年度版 囲碁名局細解
囲碁編集部編
予定価1300円

59年度「囲碁」の付録・名局細解をまとめたもの。手に汗握る熱戦を忠実に再現。全12冊箱入り。

犬のブリーディング
こんなときどうする
●繁殖の心得A～Z
永野泰著
定価980円

ブリーダーのルールからペット・ショップ、用語などを解説したブリーダーのためのルールブック。

コックピットインフォメーション
パーソナル無線情報'85 Part・2
初歩のラジオ別冊
定価480円

MSXソフト集 Part・3
初歩のラジオ別冊
定価800円

野菜の新技術と新資材
農耕と園芸別冊
定価2000円

ファミコンソフト集 001
ファミコン大作戦

わずか800円でオリジナルゲームソフトが10数本！ゲームづくりも楽しめちゃいます。
●5月10日刊　地球防衛軍編

ファミリーベーシック ホップ
ファミコン大作戦

ファミコンも使い方次第で立派なBASICマシン。800円でBASICを攻略！
●5月10日刊　地球防衛軍編

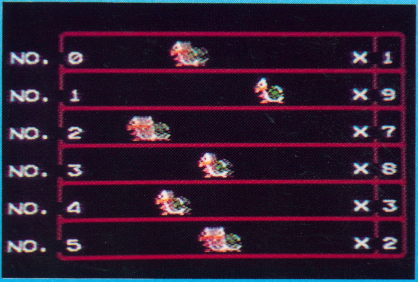
ファミコン大作戦

ファミコンソフト集

001

誠文堂新光社


HI-SCORE 0
BONUS 400
SCORE 0
PENPEN 2


NO. 0 X 1
NO. 1 X 9
NO. 2 X 7
NO. 3 X 8
NO. 4 X 3
NO. 5 X 2